TIMEPROVE: Propose, then Verify for Efficient Long Video Temporal Reasoning in Activities of Daily Living

Anonymous ACL submission

Abstract

Long Video Question Answering (LVQA) requires identifying sparse, query-relevant evidence within hours-long untrimmed videos. Existing approaches either process videos densely with large vision-language models (VLMs), incurring prohibitive computational cost, or rely on sparse caption-based reasoning, which often misses temporally localized and motion-centric evidence. We introduce **TIMEPROVE**, a cost-efficient hybrid framework for temporally grounded reasoning in long videos. TIMEPROVE first employs lightweight modules to generate action-grounded answer–evidence hypotheses and subsequently invokes an expensive VLM only for targeted verification. The core of our framework lies in the **Action-based Candidate Evidence (ACE)** module, which converts temporally localized actions into query-conditioned candidate answers and supporting evidence windows through lightweight LLM reasoning. We further introduce **OPENTSUBENCH (OTB)**, an open-ended benchmark designed to evaluate temporally grounded reasoning in real-world Activities of Daily Living (ADL) scenarios. Experiments show that TIMEPROVE outperforms the strongest baseline on OTB by 7.3%, while reducing VLM calls by 75% and inference cost by 93%. Furthermore, without explicit temporal grounding training, TIMEPROVE achieves competitive performance on CHARADES-STA, and reaches state-of-the-art results when enhanced with grounding VLMs.

1 Introduction

In long-form Activities of Daily Living (ADL) videos, the answer to a natural-language query may hinge on a few seconds of subtle visual evidence. *Taking medication, sipping water,* or *picking up a small object* can be easy to miss, and visually similar activities may differ only in fine-grained hand-object interactions. For example, answering *"Has the person taken their medicine, and did they drink water afterwards?"* requires finding the relevant moments in a long video and analyzing them closely enough to distinguish the intended actions.

This makes ADL Long Video Question Answering (LVQA) fundamentally different from short, curated action recognition benchmarks (Soomro et al., 2012; Liu et al., 2019; Shahroudy et al., 2016; Das et al., 2019; Kay et al., 2017), where ∼10-second clips usually contain a single prominent action. Accurate LVQA therefore requires both temporal search and fine-grained visual-language reasoning. Large VLMs (Zhang et al., 2025a; Reilly et al., 2025; Wang et al., 2023; Bai et al., 2025; Hurst et al., 2024) are effective for the latter, but applying them directly to hour-long videos is often impractical. For example, for a 60-minute video sampled at 1 FPS, a VLM with a SigLIP-style vision encoder (Zhai et al., 2023) at 384×384 resolution with patch size 14 produces approximately 729 visual tokens per frame. This yields $3600 \times 729 \approx 2.6 \times 10^6$ visual tokens before accounting for text prompts, timestamps, or output tokens. Thus, hour-long videos can consume more than *2 million tokens*, making full-video inference difficult to scale. In practice, such inputs may exceed context limits, while cloud-based inference further introduces latency and monetary cost (Comanici et al., 2025; Achiam et al., 2023). These constraints motivate LVQA approaches that avoid unnecessary processing of the full video while still preserving the fine-grained visual evidence needed for accurate answers.

A natural alternative to full-video VLM inference is to first caption the video (Pan et al., 2016; Chen et al., 2023; Wang et al., 2023) and perform reasoning over the resulting text, which is substantially cheaper to process than dense visual tokens (Pan et al., 2016; Chen et al., 2023; Wang et al., 2023, 2024, 2025d; Zuo et al., 2025; Ma et al., 2025; Kahatapitiya et al., 2025). However, the overall efficiency of such pipelines critically depends

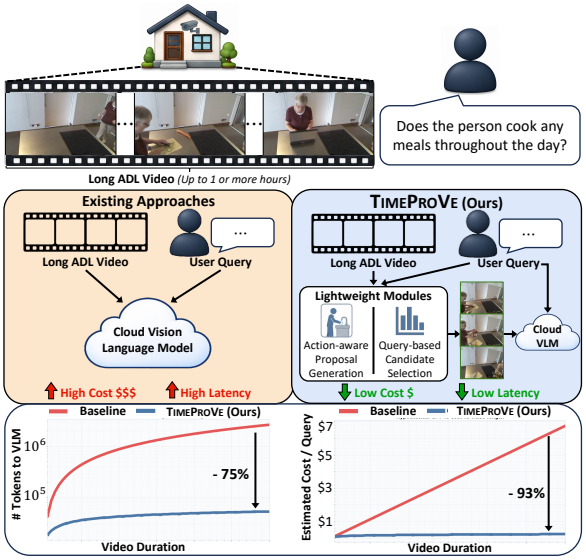

Figure 1: TIMEPROVE reduces long-video LVQA cost by proposing query-relevant evidence locally before VLM verification. Instead of processing the full video, it sends only short targeted clips to the cloud VLM.

on the captioning model itself: while lightweight captioners can reduce computation, generating captions with large VLMs may still require repeated expensive inference over long videos. Moreover, the effectiveness of such approaches depends critically on whether the generated captions preserve the query-relevant evidence required for accurate reasoning. This is especially difficult in ADL question answering, where answers may hinge on brief, fine-grained actions rather than salient frame content. Sparse captions often omit temporal grounding, subtle motion cues, hand-object interactions, or concurrent activities. Once such evidence is absent from the text, downstream LLM reasoning cannot recover it, leading to wrong responses.

Thus, we propose a different paradigm: keep *visual* reasoning instead of converting the video to text, but restrict it to the moments where it matters. We instantiate this idea with **T**ime-aware **Pro**posal and **Ve**rification (**TIMEPROVE**), a cost-efficient hybrid framework for temporal reasoning in long videos. Rather than exhaustively processing the entire video, TIMEPROVE first employs lightweight modules to identify answer-relevant evidence and subsequently invokes an expensive VLM only for targeted visual verification. As illustrated in Figure 1, TIMEPROVE consists of two components. The first is the **Action-based Candidate Evidence (ACE)** module, which efficiently processes the full video in a single pass. ACE employs a lightweight temporal action detector to identify and localize actions, yielding a sparse temporal action timeline describing what actions occur and when they occur. Then, conditioned on both the user query and this action timeline, a lightweight LLM generates candidate answers together with supporting evidence windows, ranked according to query relevance. These evidence windows may correspond to individual actions or short merged intervals when broader temporal context is required. The second component is a **Temporal Verifier**, which invokes an expensive VLM only on selected short RGB evidence clips. Given a candidate answer and its associated evidence window from ACE, the verifier determines whether sufficient visual evidence supports the hypothesis. If verified, TIMEPROVE returns both the answer and its corresponding semantic and visual evidence; otherwise, it proceeds to the next candidate. Consequently, TIMEPROVE uses VLMs for targeted verification rather than exhaustive search over long videos. In practice, TIMEPROVE naturally supports a hybrid deployment setting, where ACE operates locally on edge devices while the Temporal Verifier leverages powerful remotely hosted VLMs.

Finally, existing LVQA benchmarks (Wang et al., 2025a; Wu et al., 2024) are largely restricted to multiple-choice settings, where answer options can implicitly guide evidence selection and temporal grounding is not explicitly evaluated. Therefore, we introduce **OPENTSUBENCH (OTB)**, an open-ended LVQA benchmark designed to evaluate temporally grounded reasoning in real-world ADL scenarios. OTB requires understanding across both short atomic actions and long-horizon composite activities, making it suitable for assessing open-ended reasoning in unstructured home environments. Empirically, TIMEPROVE outperforms the strongest baseline on OTB by 7.3% while reducing VLM invocations by 75% and lowering inference cost by 93%. We further evaluate the generality of our framework on a temporal grounding task using CHARADES-STA dataset. Despite not being explicitly trained for temporal grounding, TIMEPROVE achieves performance comparable to specialized temporal grounding VLMs. Moreover, when temporal grounding VLMs are integrated within the ACE module, TIMEPROVE achieves state-of-the-art performance, demonstrating the robustness of our framework. Our key contributions are summarized below:

- We introduce **TIMEPROVE**, a novel hybrid framework that performs lightweight long-

video temporal reasoning to generate action-grounded hypotheses and verifies only sparse RGB evidence using an expensive VLM.

- We design the **Action-based Candidate Evidence (ACE)** module, the first module of its kind to transform detected actions into query-conditioned answer-evidence candidates through lightweight LLM reasoning and structured reranking.
- We introduce **OPENTSUBENCH**, an open-ended benchmark for temporally grounded LVQA in real-world untrimmed ADL videos.
- TIMEPROVE achieves a 7.3% improvement over the strongest baseline on OTB while requiring substantially fewer VLM invocations and lower inference cost. Additionally, TIMEPROVE achieves state-of-the-art performance on temporal grounding for CHARADES-STA.

2 Related Works

Vision Language Models for Long Video Understanding. Long Video VLMs face a token bottleneck, where naïvely, encoding untrimmed videos produces sequences that exceed the context window of any language model and dilute relevant evidence with irrelevant background. Existing approaches for VLMs can be categorized into three main families. First, token compression methods such as LongVLM (Weng et al., 2024), VideoChat-Flash (Li et al., 2024), Bimba (Islam et al., 2025), and STORM (Jiang et al., 2025) hierarchically merge tokens within a fixed budget. Second, frame and token selection methods recast the problem as retrieval by ranking frames by query similarity (Liang et al., 2024; Zhang et al., 2025b), combinatorial coverage (Yu et al., 2024) or adaptive policies that decide how many frames to keep (Buch et al., 2025; Tang et al., 2025). A third category builds memory mechanisms (Song et al., 2024; Diko et al., 2025) by maintaining a bounded representation that updates incrementally. However, in these methods the cost of inference scales with video length. Furthermore, token compression discards rare evidence or fine-grained details, token selection is brittle to early selection errors and may fail to capture complex temporal dependencies across multiple distant frames. Memory based representations are sensitive to memory update strategies and are prone to information drift. TimeChat (Ren et al., 2024), TimeSuite (Zeng et al., 2024) introduce temporal grounding in VLMs by training time-aware encoders. However, these methods depend on dense timestamp-aligned instruction tuning datasets for training which is expensive for long videos. Time-R1 (Liu et al., 2025) and Time-Zero (Wang et al., 2025c) attempt to relax this by training with timestamp aware rewards, but Reinforcement Learning on long videos is unstable and sample-inefficient. Complementary to these approaches, TIMEPROVE routes the dense computation to a lightweight local temporal action detection module and reserves the VLM for verification of short, query-relevant clips drawn from a sparse action prior.

Agentic Frameworks for Long Video Understanding. Recently, agent-based frameworks for long video understanding decouple the problem into perception and planning where an external LLM iteratively queries a VLM and accumulates evidence until an answer is obtained. VideoAgent (Wang et al., 2024), LangRepo (Kahatapitiya et al., 2025), VideoLucy (Zuo et al., 2025) maintain a semantic store of captions and use an LLM to extract the answer from the captions. VideoTree (Wang et al., 2025d) builds a hierarchical query-adaptive tree of candidate moments. More recent recursive grounding approaches such as RevisionLLM (Hannan et al., 2025) and AIR (Zou et al., 2025) progressively refine temporal boundaries through reason-guided iteration. However, in these systems the choice of which frames to inspect is driven by an LLM operating on sparse captions or similarity scores, hence it is detached from any learned visual prior over which actions occur. Furthermore, the feedback signal between iterations is unstructured chain-of-thought rather than a grounded residual evidence representation. In contrast to these frameworks, TIMEPROVE selects candidate windows from a learned action prior produced by an action detector. Additionally, the feedback loop is structured by an explicit residual signal to the proposal generator for calibrating windows. Unlike prior agentic systems, TIMEPROVE returns both semantic and visual evidence to expose a provenance chain.

3 Method

In long video question answering (LVQA), the goal is to answer a free-form natural language query q over an untrimmed video V of duration L. Let $V = \{f_t\}_{t=1}^{T}$ denote a sequence of T frames. A direct approach is to use a large VLM to estimate $a^\star =$

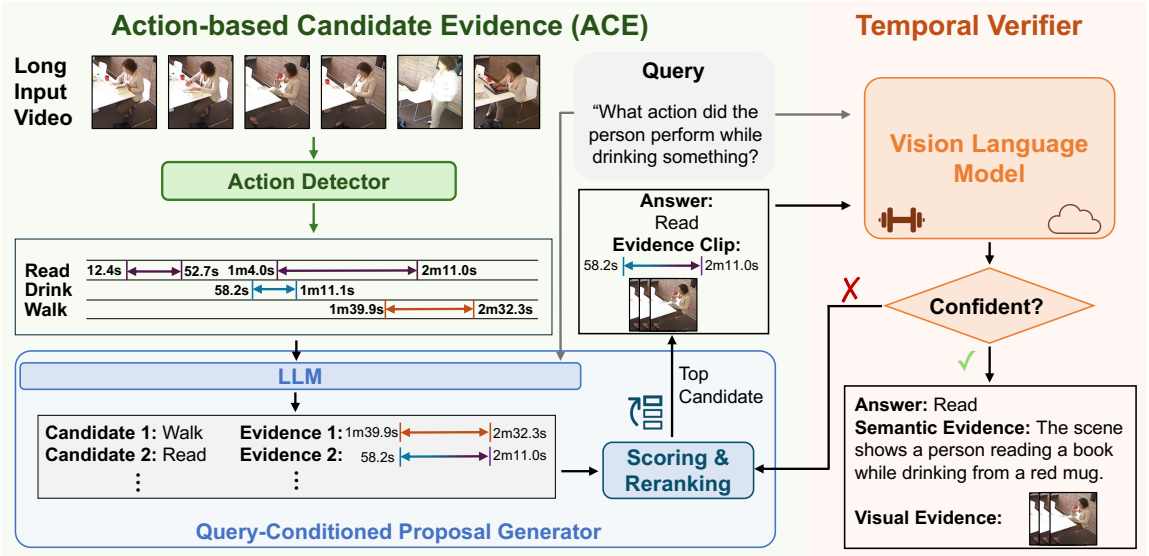

Figure 2: **Overview of TIMEPROVE.** ACE constructs a temporal action structure from the full video and proposes query-conditioned hypotheses. Only the selected short evidence clip is sent to the VLM for verification, avoiding full-video VLM inference.

$\arg\max_a\ p(a \mid q, V)$. However, applying a large VLM over an entire long video is prohibitively expensive, as the number of visual tokens scales with video duration.

Therefore, rather than processing the entire video or repeatedly invoking the VLM across many candidate temporal segments, our goal is to identify a small set of candidate windows $\mathcal{W} = \{w_k = [s_k, e_k]\}_{k=1}^{K}$, where $e_k - s_k \ll L$, and verify only the most promising evidence. Consequently, we propose the **T**ime-aware **P**roposal and **V**erification **F**ramework, TIMEPROVE, a cost-efficient framework for LVQA. As shown in Figure 2, TIMEPROVE consists of two main components: (i) an **Action-based Candidate Evidence (ACE) Module**, which operates over the full video using lightweight models to obtain query-conditioned candidate hypotheses, and (ii) a **Temporal Verifier**, which performs fine-grained verification only on short RGB clips selected by ACE. This design assigns broad temporal proposal generation to lightweight computation, reserving expensive VLM inference for targeted visual verification.

3.1 Action-based Candidate Evidence (ACE)

ACE serves as the lightweight component of TIMEPROVE. Its objective is to transform a long video into a compact temporal action representation that can be leveraged to generate candidate answer hypotheses before invoking an expensive VLM. ACE consists of two submodules: (i) **Action Detector**, which performs a single pass over the video to temporally localize actions, and (ii) **Query-conditioned Proposal Generator**, which employs an edge LLM to reason over the localized actions and produce query-conditioned candidate answers along with corresponding evidence windows.

3.1.1 Action Detector

First, we divide the given video V into T contiguous temporal segments. Following prior work in temporal action detection (Dai et al., 2022a; Sinha et al., 2026), each segment is encoded using a frozen visual backbone such as I3D (Carreira and Zisserman, 2017) or CLIP (Radford et al., 2021), producing a feature sequence $v \in \mathbb{R}^{T \times D}$, where D is the feature dimension.

The Action Detection Module predicts action probabilities over all temporal segments:

$$\mathbf{P} = f_{\text{act}}(v) \in [0,1]^{T \times |C|}, \tag{1}$$

where $|C|$ denotes the number of action classes. We threshold $\mathbf{P}$ and decode maximal contiguous activations for each class into an event timeline:

$$\mathcal{A} = \{(c_i, s_i, e_i)\}_{i=1}^{N}, \qquad 0 \le s_i < e_i \le L. \tag{2}$$

Here, c_i is the detected action label and $[s_i, e_i]$ is its temporal extent in the original video. This timeline provides a compact temporal structure over the video, indicating which actions occur and when they occur. Importantly, the full video is processed only once by the lightweight Action Detection Module. The original RGB frames are accessed again only after the Action Detector selects a short evidence window for cloud verification.

3.1.2 Query-conditioned Proposal Generator

The event timeline $\mathcal{A}$ provides a compact description of which actions occur and when they occur. However, it is not directly tied to the user query. The role of the Proposal Generator is therefore to convert this local action structure into a small set of query-conditioned hypotheses that can later be verified by the VLM.

Given the query q and the event timeline $\mathcal{A}$, an edge LLM proposes candidate answers together with supporting temporal windows:

$$\mathcal{H}_q = \{(a_i, w_i)\}_{i=1}^{M}, \tag{3}$$

where a_i denotes a candidate answer and $w_i = [s_i, e_i]$ denotes the corresponding evidence window. The Proposal Generator constructs these evidence windows at two levels of temporal granularity.

First, for every detected event $(c_i, s_i, e_i) \in \mathcal{A}$, we create an atomic evidence window, $w_i^{\text{atom}} = [s_i, e_i]$. The set of all atomic windows is denoted by $\mathcal{W}_{\text{atom}} = \{w_i^{\text{atom}}\}_{i=1}^{N}$. Atomic windows preserve the finest temporal resolution of the Action Detector and are useful when the query can be answered from a single localized action. However, many LVQA queries require context beyond a single detected event. For example, questions involving concurrent actions, object interactions, or temporal relations may require observing multiple neighboring or overlapping actions together. In order to capture such cases, the edge LLM is prompted with the query q and the event timeline $\mathcal{A}$ to identify groups of atomic windows that should be considered jointly.

Let $\mathcal{G}_q = \{G_j\}_{j=1}^{J}$ denote the groups proposed by the edge LLM, where each group $G_j \subseteq \{1, \ldots, N\}$ contains the indices of events that are relevant to the query. For each group G_j, we define a merged evidence window:

$$w_j^{\text{merge}} = \left[\min_{i \in G_j} s_i,\ \max_{i \in G_j} e_i\right]. \tag{4}$$

The set of all merged windows is denoted by $\mathcal{W}_{\text{merge}} = \{w_j^{\text{merge}}\}_{j=1}^{J}$. $\mathcal{W}_{\text{merge}}$ allows the Proposal Generator to adapt the temporal extent of the evidence to the query. For instance, for the question *"What action did the person perform while drinking something?"*, the LLM may merge the window corresponding to *drink* with nearby or overlapping actions such as *read*, producing an evidence window that contains the necessary context for verification.

The final query-conditioned candidate window set is then $\mathcal{W}_q = \mathcal{W}_{\text{atom}} \cup \mathcal{W}_{\text{merge}}$. Each window in $\mathcal{W}_q$ is paired with a candidate answer proposed by the edge LLM to form the hypothesis set $\mathcal{H}_q$. This design preserves precise localization through atomic windows while allowing the evidence window to expand only when the query requires broader temporal context.

Scoring and Reranking. The hypothesis set $\mathcal{H}_q$ contains candidate answers paired with temporal evidence windows. Although several hypotheses may be plausible, sending each corresponding RGB clip to the cloud VLM would be inefficient. Therefore, before any visual evidence is transmitted, the Query-conditioned Proposal Generator performs a local ranking step that estimates how likely each window is to contain the evidence needed to answer the query.

Let $\mathcal{A}(w)$ denote the detected events overlapping with window $w = [s_w, e_w]$. For each event a, $\mathcal{T}(a)$ denotes the normalized content-token set derived from its action label, containing lemmatized nouns and verbs after stop-word filtering. We define the window-level token set as $\mathcal{T}(w) = \bigcup_{a \in \mathcal{A}(w)} \mathcal{T}(a)$, and let $\mathcal{Q}(q)$ denote the corresponding content-token set extracted from the query. The ranking score combines four complementary criteria: whether the window occurs at a temporally plausible position, whether it contains an action strongly related to the query, whether it covers the query concepts collectively, and whether it remains compact enough for efficient verification.

Let $\tau = \tau(q)$ denote the temporal intent inferred from the query. We first compute a temporal compatibility score:

$$R_{\text{tmp}}(w, q) = \begin{cases} 1 - e_w/L, & \tau = \textsc{before}, \\ s_w/L, & \tau = \textsc{after}, \\ 1 - s_w/L, & \tau = \textsc{first}, \\ s_w/L, & \tau = \textsc{last}, \\ (e_w - s_w)/L, & \tau \in \{\textsc{between}, \textsc{state}\}, \\ 1/2, & \text{otherwise}. \end{cases} \tag{5}$$

This term acts as a soft temporal prior, biasing the ranking toward windows whose positions are compatible with the query intent.

Next, the semantic relevance score rewards the strongest action-level match within the window:

$$R_{\text{sem}}(w, q) = \frac{1}{Z_q} \max_{a \in \mathcal{A}(w)} |\mathcal{Q}(q) \cap \mathcal{T}(a)|, \tag{6}$$

where $Z_q = \max(|\mathcal{Q}(q)|, 1)$. This best-match form is useful for merged windows, since a highly

relevant action should not be penalized simply because the window also contains surrounding context. In contrast, the coverage score measures how much of the query content is represented by the window as a whole:

$$R_{\text{cov}}(w, q) = \frac{|\mathcal{Q}(q) \cap \mathcal{T}(w)|}{Z_q}. \tag{7}$$

Thus, R_{sem} favors a strong local match, while R_{cov} favors windows that jointly cover multiple query-relevant concepts. Finally, we use $R_{\text{len}}(w) = (e_w - s_w)/L$ to penalize unnecessarily long clips. The final local ranking score is:

$$\begin{aligned} R(w \mid q) = R_{\text{tmp}}(w, q) + R_{\text{sem}}(w, q) \\ + R_{\text{cov}}(w, q) - R_{\text{len}}(w). \end{aligned} \tag{8}$$

Consequentially, sorting the candidates by this score yields:

$$\mathcal{H}_q^* = \left[(a_{(1)}, w_{(1)}), \ldots, (a_{(M)}, w_{(M)})\right], \tag{9}$$

where $R(w_{(j)} \mid q) \geq R(w_{(j+1)} \mid q)$ for $j = 1, \ldots, M-1$.

This scoring function is the main mechanism that turns the raw action timeline into a query-conditioned hypothesis structure. Rather than treating all detected events as equally relevant, it organizes candidate answers by their temporal, semantic, and cost-aware compatibility with the query. Consequently, the top-ranked hypothesis serves as the most likely answer-evidence pair and is selected for expensive VLM verification.

3.2 Temporal Verifier

In TIMEPROVE, ACE efficiently narrows the search space, but the action timeline alone cannot capture all fine-grained visual details needed for answering, such as objects, attributes, interactions, and scene context. Therefore, TIMEPROVE uses a cloud VLM only as a verifier over selected short clips, rather than as a full-video reasoner.

At verification step t, let (a_t, w_t) be the highest-ranked unverified hypothesis from $\mathcal{H}_q^*$, where $w_t = [s_t, e_t]$. We extract the corresponding RGB evidence clip $\widetilde{V}_t = V[s_t, e_t]$ and send only this clip, together with the query and candidate answer, to the VLM, $(c_t, \hat{a}_t, d_t) = f_{\text{vlm}}(\widetilde{V}_t, q, a_t)$. Here, $c_t \in \{0, 1\}$ indicates whether the clip contains sufficient visual evidence, $\hat{a}_t$ is the verified answer, and d_t is the semantic evidence extracted from the clip.

If $c_t = 1$, TIMEPROVE returns:

$$(a^*, \mathcal{S}^*, \mathcal{V}^*) = (\hat{a}_t, d_t, \widetilde{V}_t), \tag{10}$$

where a^* is the final answer, $\mathcal{S}^*$ is the semantic evidence, and $\mathcal{V}^*$ is the visual evidence clip. If $c_t = 0$, the verifier rejects the hypothesis and proceeds to the next candidate in $\mathcal{H}_q^*$. The process stops when a candidate is accepted or the verification budget is exhausted.

4 OPENTSUBENCH (OTB)

LVQA is most useful when a model can not only answer a question, but also identify the temporal evidence that supports the answer. This requirement is especially important for ADL, where the relevant evidence may occupy only a few seconds within a long, visually redundant recording. Existing LVQA benchmarks often emphasize multiple-choice evaluation, report aggregate accuracy without diagnostic breakdowns, or omit precise temporal evidence. As a result, they make it difficult to evaluate whether a model is genuinely grounded or merely producing the right answer from language priors or dataset biases.

Therefore, we introduce **OPENTSUBENCH** (OTB), an open-ended, temporally grounded QA benchmark built on the Toyota Smarthome Untrimmed Dataset (TSU) (Dai et al., 2022b). OTB contains 3,567 question-answer pairs over 185 untrimmed ADL videos, with an average video duration of 21 minutes. Each question is paired with one or more supporting temporal intervals, allowing models to be evaluated both on answer correctness and on whether they localize the evidence used to answer the question.

The benchmark is constructed from timestamped TSU action annotations. We first canonicalize each video into an action timeline, instantiate a library of templated questions over the timeline, process them into natural language using a constrained LLM, and then filter them for ambiguity, triviality, and redundancy. Full construction details, prompts, filtering rules, and additional statistics are provided in the Appendix.

5 System Evaluation

5.1 Implementation Details

For the Action Detector in ACE module, we use MS-Temba (Sinha et al., 2026), a lightweight temporal action detector with 17M parameters. Videos are divided into contiguous 16-frame windows, and segment-level features are extracted using either CLIP-L/14 (Radford et al., 2021) or I3D (Carreira and Zisserman, 2017). MS-Temba is either trained on Toyota Smarthome Untrimmed (Dai

Table 1: **Comparison with State-of-the-Art on OPENTSUBENCH.**

Method	LLM	VLM	Object Centric	Temporal Positioning	Compositional Actions	State Transition	Long-Horizon Sparse Evidence	Overall
SFT-Based Frameworks								
VideoLLaMA3 (Zhang et al., 2025a)	Qwen 2	-	7.8	22.3	21.5	71.9	15.7	22.6
Qwen2.5-VL (Yang et al., 2024)	Qwen 2.5	-	72.1	40.9	26.6	27.2	35.8	39.3
VideoChat-Flash (Li et al., 2024)	InternLM2	-	63.6	36.4	29.3	66.7	29.7	37.8
VTimeLLM (Huang et al., 2024)	LLaMA-2-7B	-	55.8	27.0	32.5	65.9	29.7	33.1
TimeChat (Ren et al., 2024)	LLaMA-2-7B	-	62.8	30.7	13.1	55.7	21.2	30.4
Time-R1 (Wang et al., 2025b)	Qwen2.5VL	-	50.8	35.6	26.9	49.2	28.8	34.9
TimeSuite (Zeng et al., 2024)	Mistral-7B	-	68.2	31.4	23.9	80.9	26.7	35.4
Agentic Frameworks								
VideoTree (Wang et al., 2025d)	GPT-5.1	-	36.8	30.3	19.4	33.7	21.2	27.3
AVP (Wang et al., 2025e)	GPT-4o	GPT-4o	11.5	20.2	9.4	42.3	3.9	14.4
GPT-4o (Hurst et al., 2024)	GPT-4o	GPT-4o	27.9	27.3	9.4	67.7	11.2	23.8
Videoagent (Wang et al., 2024)	GPT-4o	GPT-4o	65.1	35.6	17.5	34.9	28.6	33.9
TIMEPROVE	Gemma4-2B	VLMA3	53.9	33.5	27.7	82.9	31.7	**37.3**
TIMEPROVE	Qwen2-7B	VLMA3	53.5	42.0	31.7	78.9	36.1	**42.7**
TIMEPROVE	Gemma4-2B	GPT-4o	49.2	47.9	37.1	70.3	35.7	**45.1**

et al., 2022b) or Charades (Sigurdsson et al., 2016), depending on the downstream benchmark, and predicts class-wise action probabilities for each temporal segment. We threshold the probabilities at $\theta = 0.5$ and decode contiguous activations into the event timeline used by ACE. For the Query-conditioned Proposal Generator, we use Gemma4-2B (DeepMind, 2026) and Qwen2-7B (Bai et al., 2023) to produce candidate answer-evidence pairs from the event timeline. For the Temporal Verifier, we evaluate VideoLLaMA3 (Zhang et al., 2025a) or GPT-4o (Hurst et al., 2024). We provide the complete prompts used for the Proposal Generator and the Temporal Verifier in the Appendix.

5.2 Comparison with State-of-the-Art

In Table 1, we compare TIMEPROVE with two families of methods, supervised fine-tuned (SFT) VLMs and agentic VLM-based frameworks. Among SFT-based methods, VideoLLaMA3 achieves the lowest performance, while temporal grounding models such as TimeChat, VTimeLLM, Time-R1, and TimeSuite improve performance on specific categories through time-aware instruction tuning but remain limited overall. Interestingly, TIMEPROVE with VLMA3 as the verifier achieves stronger overall performance. Using Gemma4-2B in ACE, TIMEPROVE achieves an improvement of 14.7% over the baseline with the same VLMA3 as Temporal Verifier. With Qwen2-7B in ACE, TIMEPROVE achieves further improvement of 5.4%, with gains across temporal positioning, compositional actions, and long-horizon sparse evidence.

Among the agentic frameworks, VideoAgent performs strongly on object-centric questions, but drops on state transitions and sparse-evidence reasoning, suggesting that frame-level captioning can capture salient objects while missing temporally specific evidence. In contrast, TIMEPROVE with GPT-4o as verifier achieves an improvement of 21.3% over the GPT-4o baseline and 11.2% over VideoAgent. Furthermore, TIMEPROVE consistently achieves a performance improvement on Temporal Positioning and Long-Horizon Sparse Evidence category, demonstrating its effectiveness in capturing temporal dependencies among actions in the long videos.

5.3 System Diagnosis

Effect of framework components. Table 2 ablates the main components of TIMEPROVE. Without the Action Detector, edge LLM, or scoring module, the system reduces to a weak caption-style or unguided baseline. Adding the Action Detector improves performance considerably, showing that the detected action timeline is a strong source of temporal structure. This validates the central assumption of our approach that grounding evidence in temporally localized actions is essential for temporal reasoning in untrimmed ADL videos. Adding the edge LLM further improves performance by 3.6%. This gain indicates that query-conditioned reasoning is necessary to identify which temporal windows should be merged to answer the query. Finally, incorporating proposal scoring and reranking achieves the best performance, indicating that heuristic-guided structuring of candidate proposals before verification is crucial for effective temporal reasoning.

Efficiency Analysis. Table 3 highlights the accuracy-efficiency tradeoff among LVQA methods. Caption-based and uniform-sampling methods process much longer video duration through repeated VLM calls, yet remain less accurate. In contrast, full-video inference has low latency but its comparable accuracy shows that exposing the VLM to more video frames does not necessarily yield better reasoning when

Table 2: **Ablating the Components of TIMEPROVE.**

Action Detection	Proposal Generator		Perf.
	Edge LLM	Score Rerank	
✗	✗	✗	24.7
✓	✗	✗	36.4
✓	✓	✗	40.0
✓	✓	✓	**42.7**

Table 3: **Efficiency comparison of TIMEPROVE with LVQA baselines.**

Method	Acc.	# Calls	Dur.	Lat.
Caption-Based	24.7	16.8	1004.8	55.0
Uniform Sampling	34.7	16.8	1004.8	27.0
Full-Video	35.0	1.0	180.0	17.6
Retrieval-Based	33.9	7.0	10.0	35.0
TIMEPROVE	44.8	8.3	123.6	18.7

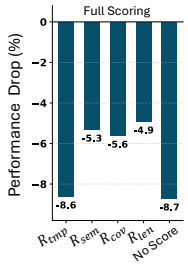

Figure 3: **Scoring Metrics**

evidence is sparse. Retrieval-based selection is efficient in processed duration, but its low performance suggests that generic retrieval often misses action-relevant evidence. In contrast, TIMEPROVE achieves the best accuracy with minimal latency overhead, because ACE narrows the search before VLM verification. This shows that the key efficiency gain is not merely reducing calls or duration in isolation, but selecting clips that are likely to contain the answer.

Table 4: **Temporal Grounding on Charades-STA**

Method	R1@0.3	R1@0.5	R1@0.7
VideoChat2 (Li et al., 2023)	9.6	3.4	1.4
TimeChat (Ren et al., 2024)	-	32.2	13.4
VTimeLLM (Huang et al., 2024)	51.0	27.5	11.4
ChatVTG (Qu et al., 2024)	52.7	33.0	15.9
TimeSuite (Zeng et al., 2024)	69.9	45.3	24.0
Time-R1 (Liu et al., 2025)	77.7	61.5	**36.8**
TIMEPROVE (+ Action Detector)	52.1	27.3	10.7
TIMEPROVE (+ TimeSuite)	71.2	50.1	24.6
TIMEPROVE (+ Time-R1)	**78.2**	**62.0**	36.1

Effect of scoring metrics. Figure 3 evaluates the drop in performance on the ablation of each term in the local ranking score. The full scoring function performs best, showing that effective evidence selection requires combining temporal, semantic, and cost-aware cues. Temporal compatibility is the most influential since removing it causes the largest drop and brings performance close to the no scoring variant. This demonstrates that for long-video temporal reasoning, it is important for the evidence window to be consistent with the temporal intent of the query. Additionally, the remaining terms contribute complementary improvements, substantiating that all components of the ranking mechanism are essential for refining the final ordering of candidate hypotheses.

5.4 Generalization to Temporal Grounding

Although TIMEPROVE is specifically designed for open-ended LVQA, its intermediate representation, i.e., localized evidence window from ACE, is inherently temporal. This makes it natural to ask whether the same mechanism can transfer to temporal grounding, where the task is to localize the interval corresponding to a language query. We evaluate this on Charades-STA in Table 4. Complementary to LVQA, Charades-STA evaluates short, free-form temporal grounding rather than open-ended reasoning over long ADL videos. Accordingly, TIMEPROVE with only the Action Detector is limited at stricter IoU thresholds, since action detectors provide event-level segments rather than the fine boundary alignment required by temporal grounding. Nevertheless, this variant remains competitive with several general video-language grounding models, suggesting that action timelines provide useful temporal structure even beyond the target LVQA setting. Notably, while strong baselines such as TimeSuite and Time-R1 achieve competitive performance on CHARADES-STA, they substantially underperform on OTB, highlighting the challenges posed by temporally grounded reasoning in long ADL videos.

Moreover, TIMEPROVE is not tied to a specific detector. When combined with stronger temporal localization backbones, it improves over TimeSuite by 1.3, 4.8, and 0.6 points respectively. With Time-R1, it further improves the looser and medium IoU thresholds while remaining comparable at the strictest threshold. These gains indicate that ACE acts as a reusable query-conditioned evidence selection layer that can plug into stronger temporal grounding modules to improve evidence localization beyond answer generation.

6 Conclusion

We propose TIMEPROVE, a cost-efficient framework for temporally grounded LVQA. Instead of processing an entire long video with a VLM, TIMEPROVE uses ACE to generate action-grounded answer–evidence hypotheses and invokes an expensive VLM only for targeted verification of short RGB clips. We further introduce OPENTSUBENCH, an open-ended benchmark for evaluating temporal reasoning in real-world ADL videos. Experiments show that TIMEPROVE improves accuracy while substantially reducing VLM calls and inference cost, and further generalizes to temporal grounding when combined with stronger temporal localization modules.

7 Limitations

TIMEPROVE assumes that the answer-relevant evidence can be captured by a small number of localized or merged action windows. This is well suited to ADL-style reasoning, but questions requiring diffuse scene understanding over very long intervals may require broader evidence aggregation. Additionally, while TIMEPROVE substantially reduces VLM usage, the final verification step still depends on the visual reasoning ability of the chosen VLM. Improving adaptive evidence aggregation, and verifier calibration are promising directions for future work.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*.

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

Shyamal Buch, Arsha Nagrani, Anurag Arnab, and Cordelia Schmid. 2025. Flexible frame selection for efficient video reasoning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 29071–29082.

Joao Carreira and Andrew Zisserman. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. In *Conference on Computer Vision and Pattern Recognition*, pages 4724–4733. IEEE.

Jun Chen, Deyao Zhu, Kilichbek Haydarov, Xiang Li, and Mohamed Elhoseiny. 2023. Video chatcaptioner: Towards enriched spatiotemporal descriptions. *ArXiv*, abs/2304.04227.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Rui Dai, Srijan Das, Kumara Kahatapitiya, Michael Ryoo, and Francois Bremond. 2022a. MS-TCT: Multi-Scale Temporal ConvTransformer for Action Detection. In *CVPR*.

Rui Dai, Srijan Das, Saurav Sharma, Luca Minciullo, Lorenzo Garattoni, Francois Bremond, and Gianpiero Francesca. 2022b. Toyota smarthome untrimmed: Real-world untrimmed videos for activity detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1.

Srijan Das, Rui Dai, Michal Koperski, Luca Minciullo, Lorenzo Garattoni, Francois Bremond, and Gianpiero Francesca. 2019. Toyota smarthome: Real-world activities of daily living. In *International Conference on Computer Vision*.

Google DeepMind. 2026. Gemma 4: Open Multimodal Models for Responsible AI Development. Google AI for Developers. Accessed: 2026-05-03.

Anxhelo Diko, Tinghuai Wang, Wassim Swaileh, Shiyan Sun, and Ioannis Patras. 2025. Rewind: Understanding long videos with instructed learnable memory. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 13734–13743.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, and 1 others. 2025. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 24108–24118.

Madeleine Grunde-McLaughlin, Ranjay Krishna, and Maneesh Agrawala. 2021. Agqa: A benchmark for compositional spatio-temporal reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11287–11297.

Tanveer Hannan, Md Mohaiminul Islam, Jindong Gu, Thomas Seidl, and Gedas Bertasius. 2025. Revisionllm: Recursive vision-language model for temporal grounding in hour-long videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19012–19022.

Bin Huang, Xin Wang, Hong Chen, Zihan Song, and Wenwu Zhu. 2024. Vtimellm: Empower llm to grasp video moments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14271–14280.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Md Mohaiminul Islam, Tushar Nagarajan, Huiyu Wang, Gedas Bertasius, and Lorenzo Torresani. 2025. Bimba: Selective-scan compression for long-range video question answering. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 29096–29107.

Jindong Jiang, Xiuyu Li, Zhijian Liu, Muyang Li, Guo Chen, Zhiqi Li, De-An Huang, Guilin Liu, Zhiding Yu, Kurt Keutzer, and 1 others. 2025. Storm: Token-efficient long video understanding for multimodal llms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5830–5841.

Kumara Kahatapitiya, Kanchana Ranasinghe, Jongwoo Park, and Michael S Ryoo. 2025. Language repository for long video understanding. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 5627–5646.

Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, and 1 others. 2017. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*.

Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, and 1 others. 2023. Mvbench: A comprehensive multi-modal video understanding benchmark. *arXiv preprint arXiv:2311.17005*.

Xinhao Li, Yi Wang, Jiashuo Yu, Xiangyu Zeng, Yuhan Zhu, Haian Huang, Jianfei Gao, Kunchang Li, Yinan He, Chenting Wang, and 1 others. 2024. Videochat-flash: Hierarchical compression for long-context video modeling. *arXiv preprint arXiv:2501.00574*.

Hao Liang, Jiapeng Li, Tianyi Bai, Xijie Huang, Linzhuang Sun, Zhengren Wang, Conghui He, Bin Cui, Chong Chen, and Wentao Zhang. 2024. Keyvideollm: Towards large-scale video keyframe selection. *arXiv preprint arXiv:2407.03104*.

Jun Liu, Amir Shahroudy, Mauricio Perez, Gang Wang, Ling-Yu Duan, and Alex C. Kot. 2019. Ntu rgb+d 120: A large-scale benchmark for 3d human activity understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Zijia Liu, Peixuan Han, Haofei Yu, Haoru Li, and Jiaxuan You. 2025. Time-r1: Towards comprehensive temporal reasoning in llms. *arXiv preprint arXiv:2505.13508*.

Ziyu Ma, Chenhui Gou, Hengcan Shi, Bin Sun, Shutao Li, Hamid Rezatofighi, and Jianfei Cai. 2025. Drvideo: Document retrieval based long video understanding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 18936–18946.

Yingwei Pan, Tao Mei, Ting Yao, Houqiang Li, and Yong Rui. 2016. Jointly modeling embedding and translation to bridge video and language. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Mengxue Qu, Xiaodong Chen, Wu Liu, Alicia Li, and Yao Zhao. 2024. Chatvtg: Video temporal grounding via chat with video dialogue large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1847–1856.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*.

Dominick Reilly, Rajatsubhra Chakraborty, Arkaprava Sinha, Manish Kumar Govind, Pu Wang, Francois Bremond, Le Xue, and Srijan Das. 2025. Llavidal: A large language vision model for daily activities of living. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24297–24308.

Shuhuai Ren, Linli Yao, Shicheng Li, Xu Sun, and Lu Hou. 2024. Timechat: A time-sensitive multimodal large language model for long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14313–14323.

Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. 2016. Ntu rgb+d: A large scale dataset for 3d human activity analysis. In *Conference on Computer Vision and Pattern Recognition*.

Gunnar A. Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. 2016. Hollywood in Homes: Crowdsourcing Data Collection for Activity Understanding. In *European Conference on Computer Vision(ECCV)*.

Arkaprava Sinha, Monish Soundar Raj, Pu Wang, Ahmed Helmy, and Srijan Das. 2026. Ms-temba: Multi-scale temporal mamba for efficient temporal action detection. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, and 1 others. 2024. Moviechat: From dense token to sparse memory for long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18221–18232.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. 2012. UCF101: A dataset of 101 human actions classes from videos in the wild. *CoRR*, abs/1212.0402.

Xi Tang, Jihao Qiu, Lingxi Xie, Yunjie Tian, Jianbin Jiao, and Qixiang Ye. 2025. Adaptive keyframe sampling for long video understanding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 29118–29128.

Weihan Wang, Zehai He, Wenyi Hong, Yean Cheng, Xiaohan Zhang, Ji Qi, Ming Ding, Xiaotao Gu, Shiyu Huang, Bin Xu, and 1 others. 2025a. Lvbench: An extreme long video understanding benchmark. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22958–22967.

Weihan Wang, Qingsong Lv, Wenmeng Yu, Wenyi Hong, Ji Qi, Yan Wang, Junhui Ji, Zhuoyi Yang, Lei Zhao, Xixuan Song, Jiazheng Xu, Bin Xu, Juanzi Li, Yuxiao Dong, Ming Ding, and Jie Tang. 2023. Cogvlm: Visual expert for pretrained language models. *ArXiv*, abs/2311.03079.

Xiaohan Wang, Yuhui Zhang, Orr Zohar, and Serena Yeung-Levy. 2024. Videoagent: Long-form video understanding with large language model as agent. In *European Conference on Computer Vision*, pages 58–76. Springer.

Ye Wang, Ziheng Wang, Boshen Xu, Yang Du, Kejun Lin, Zihan Xiao, Zihao Yue, Jianzhong Ju, Liang Zhang, Dingyi Yang, and 1 others. 2025b. Time-r1: Post-training large vision language model for temporal video grounding. *arXiv preprint arXiv:2503.13377*.

Ye Wang, Boshen Xu, Zihao Yue, Zihan Xiao, Ziheng Wang, Liang Zhang, Dingyi Yang, Wenxuan Wang, and Qin Jin. 2025c. Timezero: Temporal video grounding with reasoning-guided lvlm. *ArXiv*, abs/2503.13377.

Ziyang Wang, Shoubin Yu, Elias Stengel-Eskin, Jaehong Yoon, Feng Cheng, Gedas Bertasius, and Mohit Bansal. 2025d. Videotree: Adaptive tree-based video representation for llm reasoning on long videos. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 3272–3283.

Ziyang Wang, Honglu Zhou, Shijie Wang, Junnan Li, Caiming Xiong, Silvio Savarese, Mohit Bansal, Michael S Ryoo, and Juan Carlos Niebles. 2025e. Active video perception: Iterative evidence seeking for agentic long video understanding. *arXiv preprint arXiv:2512.05774*.

Yuetian Weng, Mingfei Han, Haoyu He, Xiaojun Chang, and Bohan Zhuang. 2024. Longvlm: Efficient long video understanding via large language models. In *European Conference on Computer Vision*, pages 453–470. Springer.

Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. 2024. Longvideobench: A benchmark for long-context interleaved video-language understanding. *Advances in Neural Information Processing Systems*, 37:28828–28857.

Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. 2021. Next-qa: Next phase of question-answering to explaining temporal actions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9777–9786.

Qwen An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxin Yang, Jingren Zhou, Junyang Lin, and 25 others. 2024. Qwen2.5 technical report. *ArXiv*, abs/2412.15115.

Sicheng Yu, Chengkai Jin, Huanyu Wang, Zhenghao Chen, Sheng Jin, Zhongrong Zuo, Xiaolei Xu, Zhenbang Sun, Bingni Zhang, Jiawei Wu, and 1 others. 2024. Frame-voyager: Learning to query frames for video large language models. *arXiv preprint arXiv:2410.03226*.

Zhou Yu, Dejing Xu, Jun Yu, Ting Yu, Zhou Zhao, Yueting Zhuang, and Dacheng Tao. 2019. Activitynet-qa: A dataset for understanding complex web videos via question answering. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 9127–9134.

Xiangyu Zeng, Kunchang Li, Chenting Wang, Xinhao Li, Tianxiang Jiang, Ziang Yan, Songze Li, Yansong Shi, Zhengrong Yue, Yi Wang, and 1 others. 2024. Timesuite: Improving mllms for long video understanding via grounded tuning. *arXiv preprint arXiv:2410.19702*.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. 2023. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11975–11986.

Boqiang Zhang, Kehan Li, Zesen Cheng, Zhiqiang Hu, Yuqian Yuan, Guanzheng Chen, Sicong Leng, Yuming Jiang, Hang Zhang, Xin Li, and 1 others. 2025a. Videollama 3: Frontier multimodal foundation models for image and video understanding. *arXiv preprint arXiv:2501.13106*.

Shaojie Zhang, Jiahui Yang, Jianqin Yin, Zhenbo Luo, and Jian Luan. 2025b. Q-frame: Query-aware frame selection and multi-resolution adaptation for video-llms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22056–22065.

Junjie Zhou, Yan Shu, Bo Zhao, Boya Wu, Zhengyang Liang, Shitao Xiao, Minghao Qin, Xi Yang, Yongping Xiong, Bo Zhang, and 1 others. 2025. Mlvu: Benchmarking multi-task long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13691–13701.

Yuanhao Zou, Shengji Jin, Andong Deng, Youpeng Zhao, Jun Wang, and Chen Chen. 2025. Air: Enabling adaptive, iterative, and reasoning-based frame selection for video question answering. *arXiv preprint arXiv:2510.04428*.

Jialong Zuo, Yongtai Deng, Lingdong Kong, Jingkang Yang, Rui Jin, Yiwei Zhang, Nong Sang, Liang Pan, Ziwei Liu, and Changxin Gao. 2025. Videolucy: Deep memory backtracking for long video understanding. *arXiv preprint arXiv:2510.12422*.

Appendix

Overview

The Supplementary material is organized as follows:

- Section A: TIMEPROVE's robustness to noisy Action Priors.
- Section B: Evidence Grounding Capability of TIMEPROVE
- Section C: OpenTSUBench (OTB): Construction and Evaluation Details
- Section E: Prompts
- Section D: Algorithmic Framework of TIMEPROVE

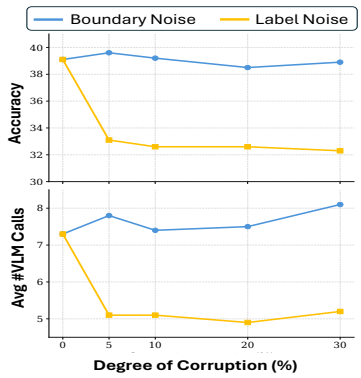

Figure 4: Effect of boundary and label noise on Accuracy and number of VLM calls.

A TIMEPROVE's robustness to noisy Action Priors

Since TIMEPROVE uses the ACE module to construct candidate evidence windows from an action timeline, it is important to understand the sensitivity of the framework to imperfect action detection. We therefore evaluate TIMEPROVE under corrupted event timelines. Starting from the detected timeline $\mathcal{A} = \{(c_i, s_i, e_i)\}_{i=1}^{N}$, we introduce two types of perturbations at increasing noise levels. For *label noise*, we randomly replace the action labels c_i of a fraction p of events with labels sampled from the remaining action vocabulary, while keeping their temporal boundaries fixed. For *boundary noise*, we perturb each event boundary as $\tilde{s}_i = s_i + \epsilon_i^s$ and $\tilde{e}_i = e_i + \epsilon_i^e$, where $\epsilon_i^s, \epsilon_i^e \sim \mathcal{N}(0, \sigma^2)$, followed by clipping to the valid video range and enforcing $\tilde{s}_i < \tilde{e}_i$. We conduct this analysis on a randomly sampled subset of 1000 QA pairs from OTB.

Figure 4 demonstrates TIMEPROVE's effectiveness in the absence of perfect action timeline. Boundary perturbations have only a limited impact up to moderate corruption levels, indicating that ACE can still propose windows that overlap with the relevant evidence even when temporal boundaries are imprecise. Label noise is more challenging because it directly affects query-conditioned proposal generation and reranking; nevertheless, the performance drop remains bounded. This is because the Temporal Verifier does not rely solely on action labels, it validates the selected RGB clip with the VLM before producing the final answer. These results show that the local action timeline acts as an efficient guide for evidence selection, while final prediction remains grounded in visual verification, making TIMEPROVE robust to imperfections in the local temporal model.

Table 5: **Evidence Grounding on OTB**

Temporal Grounding	Captioning Based	TIMEPROVE
NA	9.1	22.3

B Evidence Grounding Capability of TIMEPROVE

Table 5 evaluates the quality of the temporal evidence selected by TIMEPROVE on OTB. Although temporal-grounding methods are trained with timestamp-based instruction-tuning data, they fail to reliably localize evidence in long, untrimmed ADL videos. Captioning-based approaches provide a stronger point of comparison: they partition the video into fixed-size clips, generate timestamped clip-level captions, and aggregate them into a video-level description, allowing each answer to be traced back to the corresponding captioned clip. We report the temporal IoU between the selected evidence and the ground-truth evidence intervals, using a 10-second buffer around the reference window. TIMEPROVE achieves a tIoU of 22.3, more than twice that of the captioning-based baseline. This result shows that ACE identifies answer-relevant evidence more precisely than dense captioning, despite operating through lightweight action-grounded proposals rather than exhaustive caption generation.

Table 6: Comparison of OTB with contemporary long-video QA benchmarks. "Temporal GT" indicates presence of ground-truth supporting intervals. "Open-ended" indicates free-form answers rather than multiple choice. "Multi-interval" indicates support for answers that depend on temporally separated evidence.

Benchmark	Avg. len.	Open-ended	Temporal GT	Multi-interval	Stratified	Domain
AGQA (Grunde-McLaughlin et al., 2021)	30 s	✓	✗	✗	✗	ADL
ActivityNet-QA (Yu et al., 2019)	3 min	✓	✗	✗	✗	internet
NExT-QA (Xiao et al., 2021)	44 s	✗ (MCQ)	weak	✗	✓	social
Video-MME (Fu et al., 2025)	17 min	✗ (MCQ)	✗	✗	✓	varied
MLVU (Zhou et al., 2025)	12 min	partial	partial	✗	✓	varied
LongVideoBench (Wu et al., 2024)	12 min	✗ (MCQ)	weak	✗	✓	varied
OTB (ours)	21 min	✓	✓	✓	✓	ADL

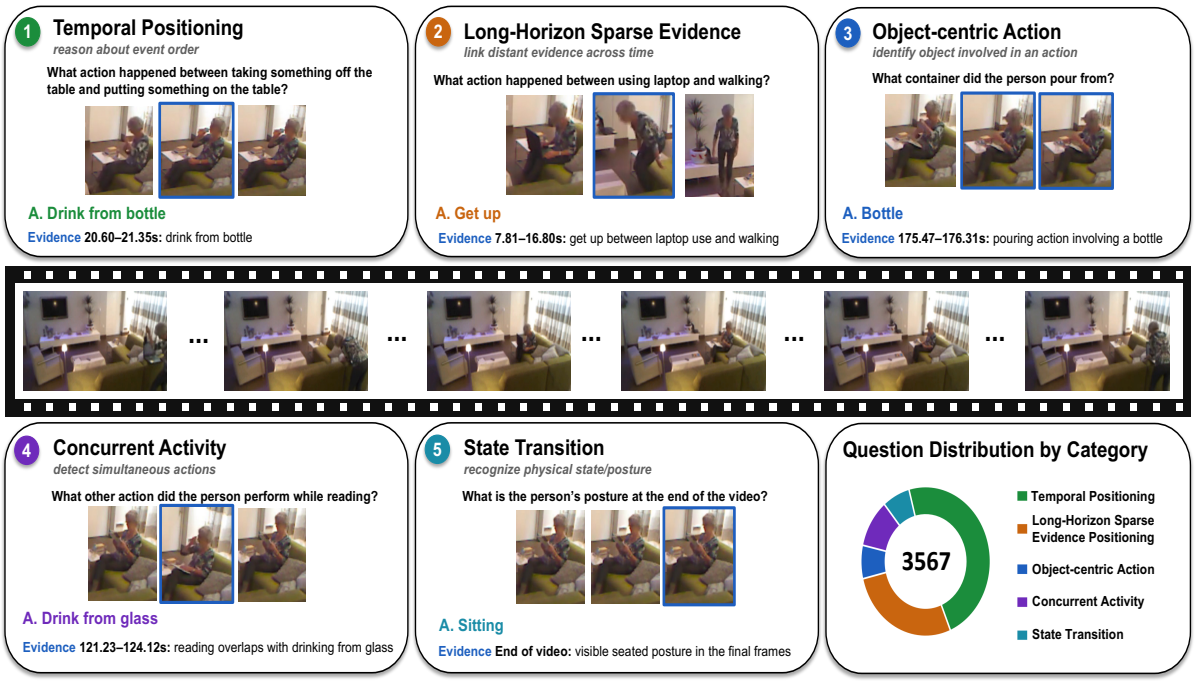

Figure 5: Overview of OPENTSUBENCH (OTB)

C OpenTSUBench (OTB): Construction and Evaluation Details

In this section, we describe **OPENTSUBENCH** (OTB), an open-ended and temporally grounded benchmark for long-horizon Activities of Daily Living (ADL) video question answering. Existing LVQA benchmarks often evaluate models through multiple-choice questions, where answer options can implicitly guide both reasoning and temporal localization. In contrast, OTB is designed to evaluate whether a model can answer free-form questions and identify the temporal evidence that supports its answer, as demonstrated in Table 6. To this end, each question in OTB is paired with one or more supporting video intervals, allowing us to evaluate both answer correctness and temporal grounding. Figure 5 illustrates OTB and its different strata.

Source Data. OTB is built on Toyota Smarthome Untrimmed (TSU) (Dai et al., 2022b), which contains untrimmed ADL videos with dense temporal annotations. The videos cover everyday activities such as preparing drinks, cooking, eating, taking pills, cleaning, posture changes, and device use. Since these activities often overlap and vary substantially in duration, TSU provides a realistic testbed for long-video temporal reasoning. We use the timestamped action annotations as the structural basis for question generation, so every question remains linked to the action event or events that justify its answer.

Benchmark Construction. We construct OTB through a pipeline, following standard benchmarks (Li et al., 2023), that preserves the temporal structure of TSU while producing natural open-ended questions.

First, we canonicalize the raw annotations into an event timeline, where each event contains an

Algorithm 1 TIMEPROVE: Local action-guided proposal and temporal verification for LVQA.

Require: Video V, question q, Action Detector f_{act}, edge LLM f_{llm}, cloud VLM f_{vlm}, verification budget M.
Ensure: Answer a^*, semantic evidence $\mathcal{S}^*$, visual evidence $\mathcal{V}^*$.

Local Action-based Candidate Evidence (ACE)

$v \leftarrow \text{EXTRACTFEATURES}(V)$
$\mathbf{P} \leftarrow f_{\text{act}}(v)$ ▷ Predict segment-level action probabilities, Eq. 1
$\mathcal{A} \leftarrow \text{DECODETIMELINE}(\mathbf{P}, \theta)$ ▷ Construct event timeline, Eq. 2
$\mathcal{W}_{\text{atom}} \leftarrow \{[s_i, e_i] : (c_i, s_i, e_i) \in \mathcal{A}\}$
$\mathcal{G}_q \leftarrow f_{\text{llm}}(q, \mathcal{A})$ ▷ Query-conditioned grouping of action windows
$\mathcal{W}_{\text{merge}} \leftarrow \text{MERGEWINDOWS}(\mathcal{G}_q, \mathcal{A})$ ▷ Eq. 4
$\mathcal{W}_q \leftarrow \mathcal{W}_{\text{atom}} \cup \mathcal{W}_{\text{merge}}$
$\mathcal{H}_q \leftarrow \text{GENERATEHYPOTHESES}(f_{\text{llm}}, q, \mathcal{A}, \mathcal{W}_q)$ ▷ $\mathcal{H}_q = \{(a_i, w_i)\}_{i=1}^{M}$
$\mathcal{H}_q^* \leftarrow \text{SORT}(\mathcal{H}_q; R(w \mid q))$ ▷ Scoring and reranking, Eq. 8

Cloud Temporal Verification

for $t = 1$ **to** $\min(M, |\mathcal{H}_q^*|)$ **do**
  $(a_t, w_t) \leftarrow \mathcal{H}_q^*[t], \quad w_t = [s_t, e_t]$
  $\widetilde{V}_t \leftarrow V[s_t, e_t]$ ▷ Extract selected RGB evidence clip
  $(c_t, \hat{a}_t, d_t) \leftarrow f_{\text{vlm}}(\widetilde{V}_t, q, a_t)$ ▷ Verify candidate answer, Eq. **??**
  if $c_t = 1$ **then**
    $\mathcal{S}^* \leftarrow d_t, \quad \mathcal{V}^* \leftarrow \widetilde{V}_t$
    return $(\hat{a}_t, \mathcal{S}^*, \mathcal{V}^*)$
  end if
end for
return $\text{FALLBACK}(\mathcal{H}_q^*)$ ▷ Return best available candidate if budget is exhausted

action label, start and end frame, temporal neighbors, and semantic tags such as `food`, `cleaning`, `posture`, `device`, and `container`. Overlapping actions are retained rather than collapsed, since concurrency is common in ADL videos and often necessary for answering questions. Next, we instantiate syntactic templates over the canonical timeline. Each template is tied to one stratum and parameterized by event indices, so the supporting interval is known at generation time.

Then we subsample the generated pool to balance template type, stratum, video, target action, and answer type. An LLM is used to formalize the retained questions while preserving event references, answer semantics, and temporal support. Finally, we filter questions with ambiguous references, trivial answers, near-duplicate wording, unreliable temporal adjacency, or paraphrases that alter meaning. To validate the QA construction, two human annotators audit the benchmark for answer correctness, temporal support, and stratum assignment.

Thus each question is paired with supporting intervals $\{[s_j, e_j]\}_{j=1}^{m_q}$. Some questions require a single localized event, while others require multiple intervals, such as verifying that one action occurs after another. These annotations are aimed to distinguish models that merely answer correctly from those that actually retrieve the relevant evidence.

D Algorithmic Framework of TIMEPROVE

Algorithm 1 demonstrates the algorithmic Framework of TIMEPROVE.

E Prompts

Action Conditioned Evidence (ACE)

"You decide which detected actions from the numbered list should be merged into one temporal evidence span to answer the question. Output a single JSON object only, no markdown fences or commentary. Schema: {\"merge_groups\": [{\"action_indices\": [int, ...], \"why\": string }]}. Each entry lists 0-based indices that belong together for answering the query; you may merge non-neighboring indices if the question calls for joint evidence. Include only groups with at least two indices; omit unrelated actions and omit singletons."

Temporal Verifier

"Write exactly three detailed sentences describing this video clip. Focus on: (1) what actions occur, especially by any people; (2) the main objects and scene elements visible; (3) how people interact with objects (handling, using, moving, or contacting them). Be concrete and specific. Use exactly three sentences.
Write exactly three sentences describing ONLY what is visible in this clip. Prefer details that help answer the question: actions, involved objects, spatial relations, and event order. If the clip does not show enough evidence for the question, still describe the clip faithfully and note that the answer cannot be determined from this clip alone. Do not provide the final answer directly; provide an evidence-focused description."

Figure 6: Prompts for ACE and Temporal Verifier in TIMEPROVE