MUSEG: Reinforcing Video Temporal Understanding via Timestamp-Aware Multi-Segment Grounding

Anonymous ACL submission

Abstract

Video temporal understanding is crucial for multimodal large language models (MLLMs) to reason over events in videos. Despite recent advances in general video understanding, current MLLMs still struggle with fine-grained temporal reasoning. While reinforcement learning (RL) has been explored to address this issue recently, existing RL approaches remain limited in effectiveness. In this work, we propose **MUSEG**, a novel RL-based method that enhances temporal understanding by introducing timestamp-aware multi-segment grounding. MUSEG enables MLLMs to align queries with multiple relevant video segments, promoting more comprehensive temporal reasoning. To facilitate effective learning, we design a customized RL training recipe with phased rewards that progressively guides the model toward temporally grounded reasoning. Extensive experiments on temporal grounding and time-sensitive video QA tasks demonstrate that MUSEG significantly outperforms existing methods and generalizes well across diverse temporal understanding scenarios.

1 Introduction

Video temporal understanding (Liu et al., 2024a; Chen et al., 2024; Cheng et al., 2025b) refers to the tasks of comprehending events based on temporal dynamics such as temporal grounding (Gao et al., 2017), dense video captioning (Wang et al., 2024), and grounded video question answering (Xiao et al., 2024). This capability is essential for multimodal large language models (MLLMs) (Hurst et al., 2024; Team et al., 2023; Bai et al., 2025) in understanding complex temporal structures in videos and making accurate, context-aware predictions or decisions based on when and how events unfold.

Despite rapid progress and impressive results in general video understanding, current MLLMs still show significant limitations in temporal understanding (Liu et al., 2024b; Li et al., 2025c). Early efforts to address this are mainly based on supervised fine-tuning (SFT) to improve temporal comprehension (Bai et al., 2025; Liu et al., 2024a; Li et al., 2025a). As reinforcement learning (RL) has been shown to significantly improve complex reasoning and comprehension in large language models (LLMs) (Guo et al., 2025), recent studies have extended RL techniques to the video domain (Feng et al., 2025; Li et al., 2025b; Wang et al., 2025; Zhang et al., 2025), encouraging models to "reason before answering". This typically involves designing a format reward to ensure a structured reasoning process and an answer reward such as Intersection over Union (IoU) to measure the correctness of the predictions.

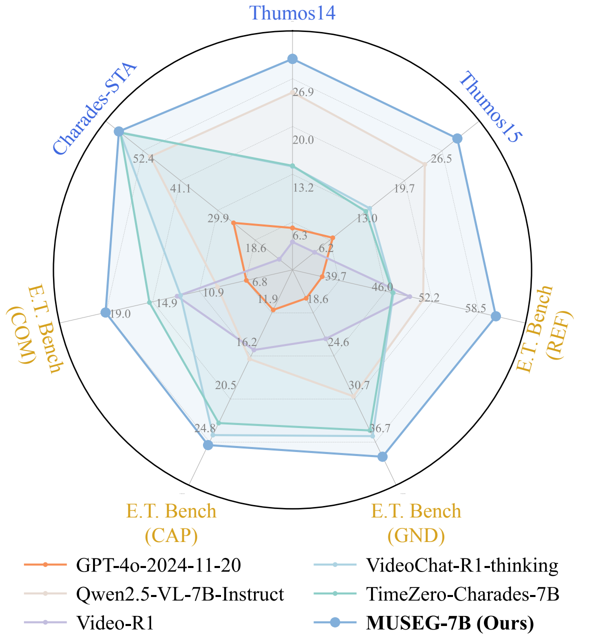
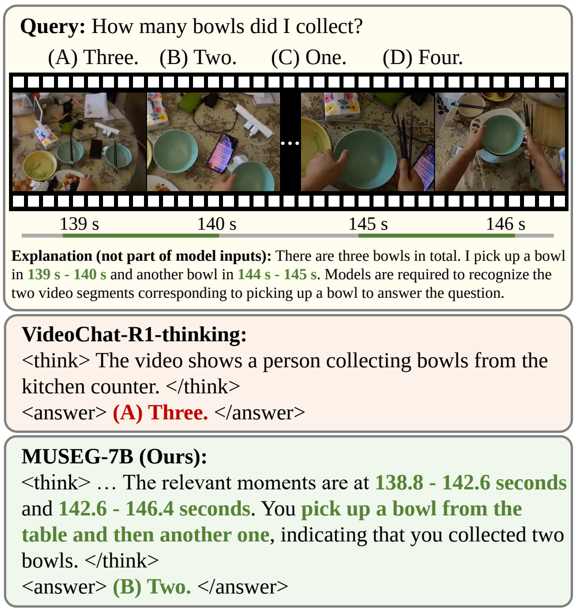


Figure 1: Performance of MUSEG-7B on various temporal grounding (Charades-STA, THUMOS14 and THUMOS15) and broader time-sensitive video understanding (E.T. Bench Subset) tasks.

However, directly applying RL to video temporal understanding tasks has not achieved the same level of performance improvement as in textual domains (Feng et al., 2025; Li et al., 2025b). We



Query: How many bowls did I collect?

(A) Three. (B) Two. (C) One. (D) Four.

Explanation (not part of model inputs): There are three bowls in total. I pick up a bowl in **139 s - 140 s** and another bowl in **144 s - 145 s**. Models are required to recognize the two video segments corresponding to picking up a bowl to answer the question.

VideoChat-R1-thinking:
<think> The video shows a person collecting bowls from the kitchen counter. </think>
<answer> **(A) Three.** </answer>

MUSEG-7B (Ours):
<think> … The relevant moments are at **138.8 - 142.6 seconds** and **142.6 - 146.4 seconds**. You **pick up a bowl from the table and then another one**, indicating that you collected two bowls. </think>
<answer> **(B) Two.** </answer>

Figure 2: An example comparing our MUSEG-7B with previous models. MUSEG-7B performs more precise, timestamp-aware reasoning by leveraging multiple key temporal cues to derive the correct answer.

attribute this limitation to two key challenges. First, most existing methods (Li et al., 2025b; Wang et al., 2025) rely solely on single-segment temporal grounding, where each input query corresponds to only one video segment. This limits the ability to capture fine-grained, multi-segment temporal information, which is essential for complex video understanding tasks. Second, although temporal understanding depends fundamentally on reasoning over temporal cues, current RL approaches often fail to model them effectively. As illustrated in Figure 2, their reasoning typically consists of brief descriptions of video content, lacking detailed temporal analysis of key events. Therefore, we argue that advancing MLLMs in video temporal understanding requires rethinking both the *training task design* and the *RL training recipe*.

In this paper, we propose timestamp-aware **MU**lti-**SE**gment **G**rounding (MUSEG), an RL-based method designed to enhance the temporal understanding and reasoning capabilities of MLLMs. On the task side, we incorporate *multi-segment grounding* into the training process, enabling models to learn from queries that align with multiple relevant video segments. This promotes stronger temporal understanding and better generalization to a wide range of time-sensitive tasks. On the training side, we introduce a customized RL training recipe with phased rewards, which progressively encourage the model to establish temporally grounded reasoning processes. Our recipe features a dedicated segment matching reward and a timestamp reward, encouraging models to perform fine-grained temporal reasoning over multiple segments as shown in Figure 2. Additionally, we employ a multi-phase training strategy that balances guided learning and exploration, ultimately achieving optimal performance. As illustrated in Figure 1, MUSEG achieves significant improvements on temporal grounding benchmarks and generalizes effectively to other time-sensitive video understanding tasks. Our contributions can be summarized as follows:

- We propose MUSEG, a novel RL-based method for video temporal understanding that enables MLLMs to reason over multiple temporally distributed events by incorporating multi-segment grounding into training.
- We design a tailored RL training recipe featuring novel reward functions and a multi-phase training strategy, effectively promoting fine-grained and temporally grounded reasoning.
- We conduct extensive experiments and analyses, showing that MUSEG consistently outperforms existing methods on video temporal understanding benchmarks, and validating the effectiveness of our task and training designs.

2 Related Work

2.1 Video Temporal Understanding

Previous research on video temporal understanding focuses on cross-references and alignments between videos and texts (Arnab et al., 2021; Luo et al., 2021; Liu et al., 2021; Xu et al., 2021; Wang et al., 2021). Recent advances in video temporal understanding have moved from these cross-modal attention-based local feature matching approaches to broader time-sensitive tasks, such as temporal grounding (Gao et al., 2017), dense video captioning (Wang et al., 2024), and grounded video question answering (Xiao et al., 2024). These methods attempt to fuse video temporal features and text features with LLMs to enhance model performance (Liu et al., 2024a; Li et al., 2025c; Yan et al., 2025).

However, these models remain suboptimal performance on temporal understanding tasks, and struggle to generalize to complex scenarios. Recent benchmarks (Liu et al., 2024a; Chen et al., 2024; Cai et al., 2024; Huang et al., 2024) highlight the gap between MLLMs and humans and the critical need for improving model abilities of temporal understanding.

2.2 RL for Video Understanding

RL has been widely adopted in various textual tasks (Shao et al., 2024; Ouyang et al., 2022; Schulman et al., 2017). Recent works apply RL to general video question answering tasks (Feng et al., 2025; Chen et al., 2025) and temporal grounding tasks (Li et al., 2025b; Cheng et al., 2025a). However, they still struggle on complex temporal grounding tasks, and there is still room for improvement in generalizing to broader temporal understanding scenarios.

3 Preliminaries: Reward Design in GRPO

Group Relative Policy Optimization (GRPO) (Shao et al., 2024) is a RL-based training method that has been widely adopted to improve reasoning abilities of LLMs. For a query, a group of responses are generated. Then, rewards $\{r_i\}$ are assigned to responses. Rewards guide optimization of the policy model, and affect model performance.

Many recent works leverage rule-based rewards in training. Deepseek-R1 (Guo et al., 2025) reaches superior results on textual tasks by training on math and code tasks with two rule-based rewards:

- **Accuracy Rewards**: Measuring whether models provide right answers to math problems or codes that can pass coding problems by verifiers or compilers.
- **Format Rewards**: Examining whether model responses are in “<think> ... </think><answer> ... <answer>” format.

4 Method

In this section, we elaborately introduce our proposed GRPO-based method **MUSEG**. It leverages multi-segment grounding as the training task, which will be detailed in Section 4.1. Followed by our designed rewards, segment matching reward and timestamp reward, in Section 4.2. Finally, we will describe our new training recipe with phased rewards in Section 4.3.

4.1 Multi-Segment Grounding Task

Temporal grounding is the task that requires models to match text queries with corresponding video segments, which helps improve temporal understanding abilities of MLLMs (Liu et al., 2024a). It includes two types of queries. The first type requires model to output a single segment corresponding to the text. We call it single-segment grounding. The other type do not specify number of segments models should output in the query, and groundtruths may be one or more segments. We call it multi-segment grounding.

Query Type	**w/ Shortcut**	**Total**
Single-Segment	15	50
Multi-Segment	4	50

Table 1: Results of preliminary empirical study. We sample single-segment grounding and multi-segment grounding queries from E.T. Bench (Liu et al., 2024a), and examine whether they can be answered by shortcut of recognizing key objects.

Single-segment grounding is widely taken as training task by previous RL-based works (Li et al., 2025b; Wang et al., 2025). However, our preliminary empirical study shows that a notable portion of single-segment grounding questions can be answered by shortcuts, for example, detecting key objects instead of understanding temporal information about events. We sample 50 questions of single-segment grounding from E.T. Bench (Liu et al., 2024a), and find that 30% of them can be answered correctly through detecting objects related to queries, as shown in Table 1. Therefore, we believe that, to improve temporal understanding abilities of MLLMs, single-segment grounding tasks are not enough.

In contrast, multi-segment grounding queries are difficult to be answered by shortcuts, as shown in Table 1. Thus, we add them to our training process. We ensure the number of single-segment grounding and multi-segment grounding queries are balanced, and our selected data are diverse in scenarios.

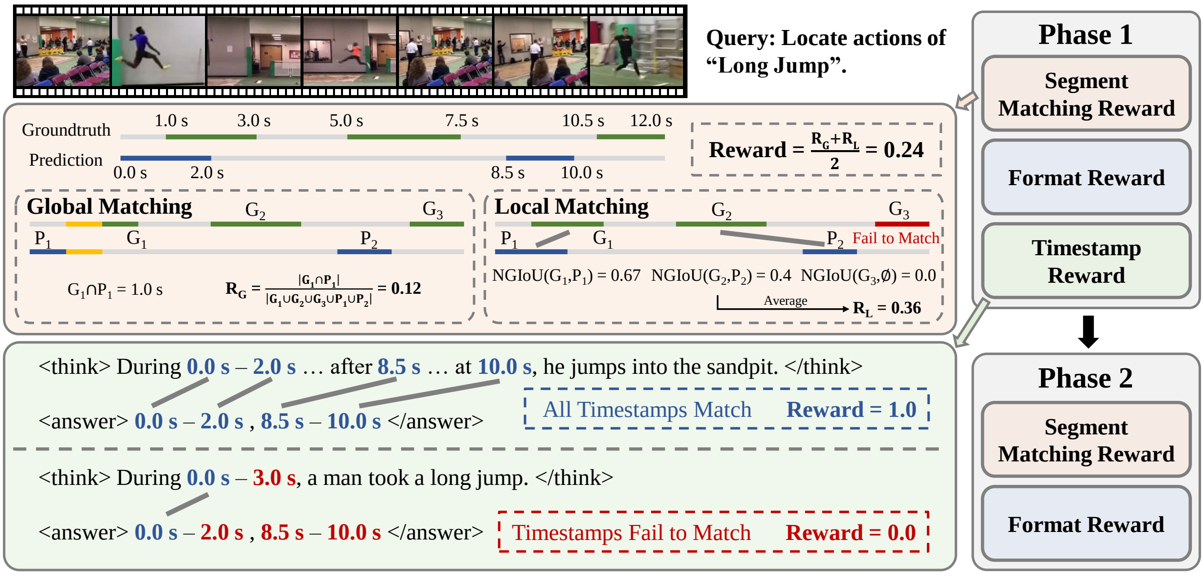
4.2 Reward Design

4.2.1 Segment Matching Reward

Segment matching reward is designed to align model outputs with groundtruths. It consists of two parts, global matching and local matching, to enhance model abilities of understanding overall video contents, and grasping detailed events, respectively.

Global matching is shown in upper left area of Figure 3 (a). We measure the overlap ratio among all the groundtruth segments $\{G_i\}$ and predicted segments $\{P_j\}$:

$$r_{\mathrm{G}} = \frac{\sum_{i,j} |G_i \cap P_j|}{|(\cup_i G_i) \cup (\cup_j P_j)|} \tag{1}$$



(a) (b)

Figure 3: Overview of MUSEG. (a) Our proposed segment matching reward (up) and timestamp reward (down). (b) RL-based training process with phased rewards of MUSEG.

In the local matching process, we pair groundtruths and predictions one-to-one as $\{(G_n, P_n)\}_{n=1}^{N}$, where $N = \max(|\{G_i\}|, |\{P_j\}|)$. As shown in upper right area of Figure 3 (a), we sort $\{G_i\}$ and $\{P_j\}$ according to their start timestamps, and match G_k with P_k, where $1 \leq k \leq \min(|\{G_i\}|, |\{P_j\}|)$. For the rest of groundtruths or predictions, we match them with empty segments ϕ. We also explore other matching strategies in Section 6.1. After matching, we assess each prediction P_n according to its paired groundtruth G_n. We leverage GIoU (Rezatofighi et al., 2019) instead of IoU for the evaluation, which better guides model optimization when the predicted video segment does not overlap with the groundtruth. We calculate NGIoU, normalized GIoU whose value is between 0 to 1:

$$\mathrm{NGIoU} = \frac{1}{2}\left(1 + \frac{|G_n \cap P_n|}{|G_n \cup P_n|} - \frac{|\mathcal{C} \backslash (G_n \cup P_n)|}{|\mathcal{C}|}\right) \tag{2}$$

where $\mathcal{C}$ is the shortest video segment covering G_n and P_n. To encourage model outputs to be closer to groundtruths, we impose a penalty when the number of groundtruth segments does not match the number of predicted segments. We define that for any G or P:

$$\mathrm{NGIoU}(G, \phi) = \mathrm{NGIoU}(\phi, P) = 0 \tag{3}$$

Finally, we calculate average NGIoU of all pairs:

$$r_{\mathrm{L}} = \frac{\sum_{n=1}^{N} \mathrm{NGIoU}(G_n, P_n)}{N} \tag{4}$$

And the final segment matching reward is:

$$r_{\mathrm{M}} = \frac{r_{\mathrm{G}} + r_{\mathrm{L}}}{2} \tag{5}$$

4.2.2 Timestamp Reward

Previous works (Feng et al., 2025; Yu et al., 2025) reveal the importance of explicitly include temporal information in reasoning process in video comprehension. Unfortunately, how to stimulate model ability of temporal-aware reasoning remains a challenging problem.

To tackle this problem, we design the timestamp reward r_{T} to enforce models to include timestamps which occur in the final answers in their thinking processes. Suppose $\{T_{\mathrm{A}}^{i}\}$ and $\{T_{\mathrm{R}}^{i}\}$ are timestamps occurring in the answer and reasoning process of a model output, then

$$r_{\mathrm{T}} = I_{\{T_{\mathrm{R}}^{i}\} \subset \{T_{\mathrm{A}}^{i}\}} \tag{6}$$

where I is indicator function. As shown in lower part of Figure 3 (a), when all the timestamps occurring in the answer are found in thinking process, models get the reward. If some timestamps fails to match, the reward is set zero. Through the timestamp reward, we encourage models to focus on temporal details during reasoning instead of thinking purely based on overall video contents.

4.3 Training Recipe with Phased Rewards

Our GRPO training process involves three rewards in total. Besides two newly designed rewards introduced in Section 4.2, format reward is also leveraged following DeepSeek-R1 (Guo et al., 2025), enforcing models to output their thinking processes and final answers in format "`<think>...</think><answer>...</answer>`":

Model	**In Domain**				**Out of Domain**									
	Charades-STA (Single-Seg)	**THUMOS14** (Multi-Seg)	**THUMOS15** (Multi-Seg)	**Perception Test** (Multi-Seg)	**E.T. Bench**					**E.T. Bench (Subset)**				
					REF	**GND**	**CAP**	**COM**	**AVG**	**REF**	**GND**	**CAP**	**COM**	**AVG**
API-based Models														
GPT-4o	25.1	5.5	6.7	-	-	-	-	-	-	37.4	16.5	11.6	6.8	18.1
Open-source ~7B Models														
Qwen2.5-VL-7B	50.2	24.9	23.4	25.3	53.1	30.7	16.2	11.3	27.8	51.0	30.3	16.5	9.3	26.8
Qwen2.5-VL-7B+SFT	28.1	15.5	15.6	20.3	24.3	11.3	15.3	6.6	14.4	27.8	12.6	15.0	8.7	16.0
E.T. Chat	45.6	23.7	24.9	9.2	38.4*	**38.0***	16.7*	13.5*	26.7	31.8*	33.8*	17.1*	11.1*	23.5
TRACE-7B	29.9*	-	-	-	33.6*	33.8*	20.3*	**25.8***	28.4	-	-	-	-	-
Video-R1	11.3	3.5	3.4	5.7	50.3	25.3	15.6	12.4	25.9	49.2	22.2	15.6	12.8	25.0
VideoChat-R1	59.4	14.3	13.4	27.1	55.8	35.6	22.1	19.5	33.3	47.0	35.9	24.1	12.5	29.9
TimeZero	59.2	14.4	12.7	26.8	55.9	35.8	21.4	17.1	32.6	46.9	35.1	22.9	15.2	30.0
MUSEG-7B (Ours)	**59.7**	**29.7**	**29.3**	**31.7**	**61.9**	37.5	**23.7**	24.0	**36.8**	**60.8**	**38.8**	**25.1**	**19.0**	**35.9**
Open-source ~3B Models														
Qwen2.5-VL-3B	41.4	12.6	12.8	19.4	51.7	20.4	13.6	8.0	23.4	52.9	20.4	12.7	7.6	23.4
TEMPURA	44.5	8.7	12.1	20.7	46.3	26.1	14.4	**10.2**	24.3	**56.4**	22.8	13.3	3.5	24.0
MUSEG-3B (Ours)	**53.7**	**21.0**	**20.3**	**29.1**	**53.9**	**30.0**	**18.7**	8.8	**27.9**	54.3	**28.7**	**18.3**	**11.8**	**28.3**

Table 2: Results of MLLMs on in-domain and out-of-domain tasks. *Results are copied from original paper. Detailed model versions are as followings: GPT-4o: GPT-4o-2024-11-20; Qwen2.5-VL-7B: Qwen2.5-VL-7B-Instruct; Qwen2.5-VL-3B: Qwen2.5-VL-3B-Instruct. VideoChat-R1: VideoChat-R1-thinking; TimeZero: TimeZero-Charades-7B.

$$r_{\mathrm{F}} = \begin{cases} 1, \text{if } o_i \text{ has right format} \\ 0, \text{otherwise} \end{cases} \tag{7}$$

Though the combination of these rewards is expected to assist models to establish temporally grounded reasoning process, we still believe that there is still room for models to find better reasoning patterns. Thus, we adopt a training recipe with phased rewards, as shown in Figure 3 (b). In the early training steps, we guide models to refer to specific timestamps in their reasoning processes. We include segment matching reward, timestamp reward, and format reward:

$$r_1 = \alpha r_{\mathrm{M}} + [\beta r_{\mathrm{T}} + (1-\beta) r_{\mathrm{F}}] \tag{8}$$

In the latter training steps, we encourage models to freely explore better forms of reasoning. Thus, we remove timestamp reward, only keeping segment matching reward and format reward:

$$r_2 = \alpha r_{\mathrm{M}} + r_{\mathrm{F}} \tag{9}$$

Through the training process with phased rewards, we achieve greater performance enhancement than solely using r_1 or r_2 for the whole training. More analyses can be found in Section 6.2.

5 Experiments

5.1 Implementations

Our training dataset is constructed from E.T. Instruct 164k (Liu et al., 2024a) and Charades-STA (Gao et al., 2017). For E.T. Instruct 164k, we only sample data from temporal video grounding (TVG) and temporal action localization (TAL) tasks. Our final training dataset consists of 12.6k samples. There are 6967 samples with a single segment, and 5633 samples with more than one segments as groundtruths.

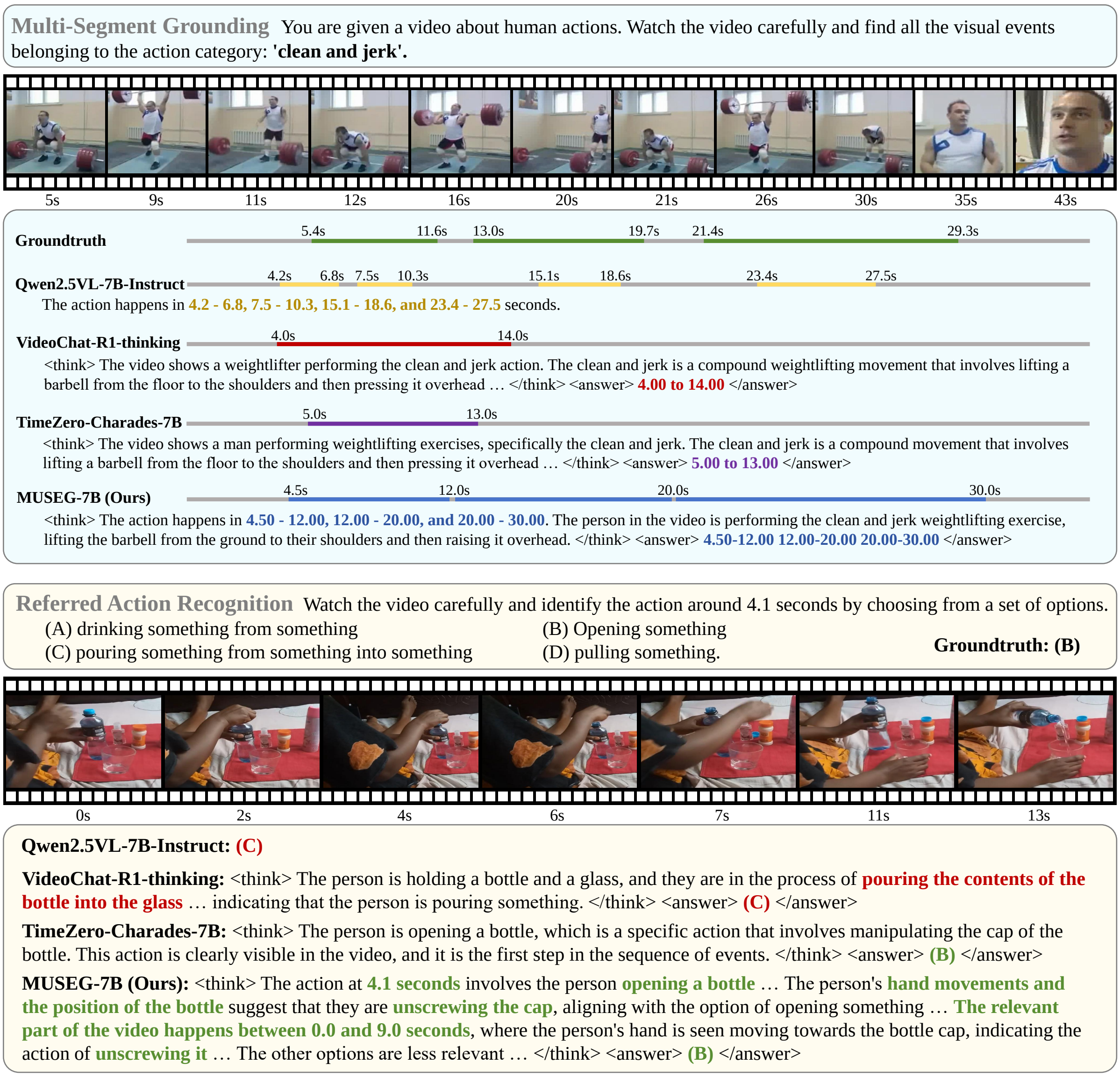
We train MUSEG-7B and MUSEG-3B based on 7B and 3B versions of Qwen2.5-VL (Bai et al., 2025). They are trained with timestamp reward for 400 steps and without timestamp reward for another 500 steps. We also conduct SFT experiments on Qwen2.5-VL-7B-Instruct with our constructed dataset. Training details can be found in Appendix A.

5.2 Baselines

We include SFT-based models and RL-based models as our baselines. For SFT-based models, we include E.T. Chat (Liu et al., 2024a), TRACE (Guo et al., 2024), and TEMPURA (Cheng et al., 2025a). For RL-based models, we include Video-R1 (Feng et al., 2025), VideoChat-R1 (Li et al., 2025b), and TimeZero (Wang et al., 2025). We also report performance of GPT-4o (Hurst et al., 2024) for reference. In consideration of inference costs, we do not report results of GPT-4o on Perception Test and the whole set of E.T. Bench. Only results on a subset of 470 samples of E.T. Bench, specified by the original paper, are reported. Detailed introductions of our baselines can be found in Appendix B.

5.3 Benchmarks and Evaluation Metrics

We evaluate MUSEG-7B and MUSEG-3B on grounding tasks (in domain) and broader time-related tasks (out of domain). We use the test set of Charades-STA (Gao et al., 2017) for single-segment grounding, and mIoU as evaluation metric. We use the validation set of THUMOS14, THUMOS15 (Idrees et al., 2017) and Perception



Multi-Segment Grounding You are given a video about human actions. Watch the video carefully and find all the visual events belonging to the action category: **'clean and jerk'**.

The action happens in 4.2 - 6.8, 7.5 - 10.3, 15.1 - 18.6, and 23.4 - 27.5 seconds.

<think> The video shows a weightlifter performing the clean and jerk action. The clean and jerk is a compound weightlifting movement that involves lifting a barbell from the floor to the shoulders and then pressing it overhead … </think> <answer> 4.00 to 14.00 </answer>

<think> The video shows a man performing weightlifting exercises, specifically the clean and jerk. The clean and jerk is a compound movement that involves lifting a barbell from the floor to the shoulders and then pressing it overhead … </think> <answer> 5.00 to 13.00 </answer>

<think> The action happens in 4.50 - 12.00, 12.00 - 20.00, and 20.00 - 30.00. The person in the video is performing the clean and jerk weightlifting exercise, lifting the barbell from the ground to their shoulders and then raising it overhead. </think> <answer> 4.50-12.00 12.00-20.00 20.00-30.00 </answer>

Referred Action Recognition Watch the video carefully and identify the action around 4.1 seconds by choosing from a set of options.

(A) drinking something from something
(B) Opening something
(C) pouring something from something into something
(D) pulling something.

Groundtruth: (B)

Qwen2.5VL-7B-Instruct: (C)

VideoChat-R1-thinking: <think> The person is holding a bottle and a glass, and they are in the process of **pouring the contents of the bottle into the glass** … indicating that the person is pouring something. </think> <answer> **(C)** </answer>

TimeZero-Charades-7B: <think> The person is opening a bottle, which is a specific action that involves manipulating the cap of the bottle. This action is clearly visible in the video, and it is the first step in the sequence of events. </think> <answer> (B) </answer>

MUSEG-7B (Ours): <think> The action at **4.1 seconds** involves the person **opening a bottle** … The person's **hand movements and the position of the bottle** suggest that they are **unscrewing the cap**, aligning with the option of opening something … **The relevant part of the video happens between 0.0 and 9.0 seconds**, where the person's hand is seen moving towards the bottle cap, indicating the action of **unscrewing it** … The other options are less relevant … </think> <answer> **(B)** </answer>

Figure 4: Cases of MUSEG-7B on multi-segment grounding (in domain) and referred action recognition (out of domain) tasks.

Test (Patraucean et al., 2023) for multi-segment grounding, and report F1 scores averaged among IoU thresholds at four levels (0.1, 0.3, 0.5, and 0.7) following Liu et al. (2024a). We evaluate model generalization with various time-related tasks in E.T. Bench (Liu et al., 2024a), including referring (REF), grounding (GND), dense captioning (CAP), and complex understanding (COM). For these tasks, we follow metrics of the original paper: accuracy for referring, F1 score for grounding, sentence similarity for dense captioning, and recall for complex understanding tasks.

5.4 Main Results

As shown in Table 2, MUSEG-7B and MUSEG-3B outperform other methods using SFT- or RL-based methods on most in-domain and out-of-domain tasks among all ~7B and ~3B models, and even surpass GPT-4o. Our method shows a significant advantage over base models. MUSEG-7B achieves more than 10% performance enhancement on all the tasks compared to its base model Qwen2.5-VL-7B-Instruct. And it is worth noting that our model gets doubled performance on complex understanding task, showing strong ability of generalization.

Video-R1 (Feng et al., 2025) does not include time-sensitive tasks in its training process, resulting in a suboptimal performance on temporal understanding tasks. Although VideoChat-R1 (Li et al., 2025b) and TimeZero (Wang et al., 2025) are trained with single-segment grounding tasks, yielding comparable single-segment grounding performance with ours, they lag behind MUSEG-7B on multi-segment grounding and other out-of-domain

Local Matching Strategy	**Charades-STA**	**THUMOS14**	**THUMOS15**	**E.T. Bench (Subset)**			
				REF	**GND**	**CAP**	**COM**
w/o Local Matching	54.7	21.2	21.4	60.9	37.2	22.9	20.8
w/ Local Matching (Sequential)	57.0	27.7	26.6	59.1	37.4	23.8	19.9
w/ Local Matching (Maximum)	55.2	25.6	25.5	54.5	36.6	21.7	15.8

Table 3: Results with different matching strategies. For all the experiments, we train Qwen2.5-VL-7B with segment matching reward, format reward and timestamp reward for 400 steps.

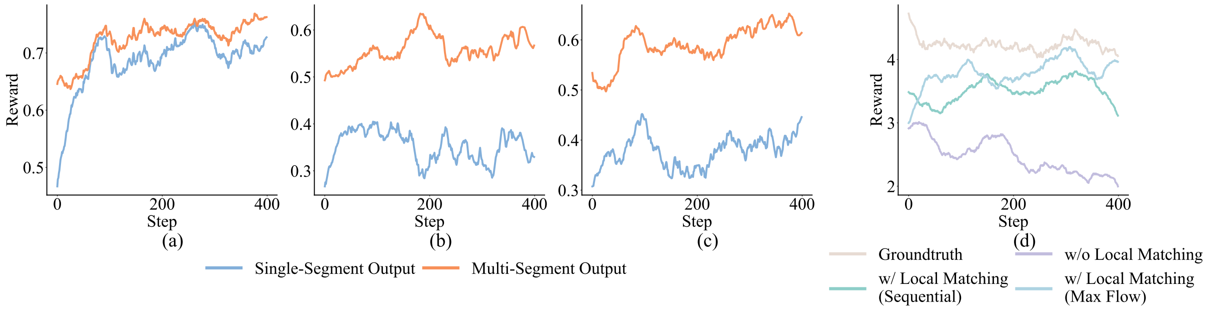


Figure 5: Segment matching reward (a) w/o local matching, (b) w/ local matching (sequential), and (c) w/ local matching (maximum). (d) Evolution of numbers of predicted segments during training process. For all the plots, we only consider queries whose groundtruths are more than one segments.

tasks. This highlights the importance of incorporating multi-segment grounding into training tasks to boost performance in time-sensitive scenarios.

5.5 Case Study

We show two cases to further demonstrate our model performance in Figure 4.

The first case is a multi-segment grounding task (in domain) with query “clean and jerk”. VideoChat-R1-thinking and TimeZero-Charades-7B only recognize the video segment corresponding to the first attempt, consistent with the fact that they are trained only with single-segment grounding tasks. In contrast, MUSEG-7B accurately localizes all three weight-lifting attempts. The performance gap highlights effectiveness of multi-segment grounding training tasks.

The second case involves referred action recognition (out of domain) query about event happening around 4.1 seconds. Seen from the video, the person first opens the bottle, then pouring water out from it. VideoChat-R1 incorrectly aligns the event of pouring water from the bottle (occurring at 11 seconds) with a 4.1-second timestamp, demonstrating a temporal misalignment in its reasoning. TimeZero-Charades-7B provides the correct answer but lacks precise timestamp references in its explanation. In contrast, MUSEG-7B exhibits superior temporal reasoning capability: it not only identifies the bottle-opening action around 4.1 seconds but also accurately localizes the corresponding video segment.

6 Analyses

6.1 Local Matching Strategies

We delve deeper to verify effectiveness of local matching in segment matching reward. We conduct experiment of removing local matching, only keeping global matching in training. Additionally, we explore another design, which involves matching groundtruths and predictions to maximize average overlap of each pair. We do this by calculating maximum weighted matching in bipartite graph. For groundtruth segments $\{G_i\}$ and predicted segments $\{P_j\}$, we construct a complete bipartite graph $\mathcal{G}$:

$$\begin{aligned} &\mathcal{G} = (\{G_i\}, \{P_j\}, E), \text{where} \\ &E = \{\text{NGIoU}(g, p) | g \in \{G_i\}, p \in \{P_j\}\} \end{aligned} \tag{10}$$

then we calculate r_L as follows:

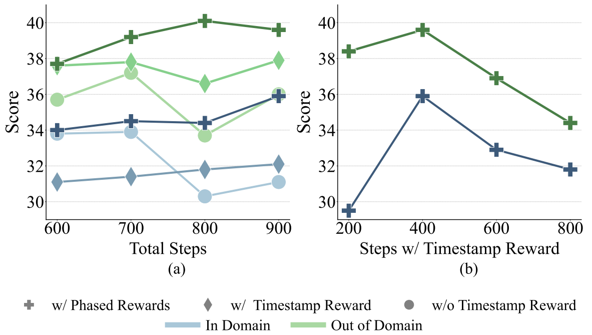
$$r_{\text{L}} = \frac{\text{Matching}(\mathcal{G})}{\max(|\{G_i\}|, |\{P_j\}|)} \tag{11}$$

where $\text{Matching}(\cdot)$ is the maximum weighted matching function. Table 3 shows that including local matching boost overall model performance compared to only keeping global matching. Additionally, sequential matching reaches better performance than maximum matching, so we finally adopt sequential matching in MUSEG.

We also notice that drops of model performance on multi-segment grounding are much larger than single-segment grounding when local matching is removed. To better understand its reason, we examine differences in rewards model would get when

Training Paradigms	Charades-STA	THUMOS14	THUMOS15	E.T. Bench (Subset)			
				REF	GND	CAP	COM
w/o Timestamp Reward	56.9	28.4	28.3	55.1	37.6	22.3	13.2
w/ Timestamp Reward	57.3	26.1	24.6	57.3	28.9	22.0	16.1
w/ Timestamp Reward for 400 Steps	59.7	29.7	29.3	60.8	38.8	25.1	19.0

Table 4: Results with different training recipes.



(a)

(b)

Figure 6: (a) Model performance with different training recipes. For the setting of phased rewards recipe, we train models with timestamp reward for 300 steps when total steps are 600 and 700, for 400 steps when total steps are 800 and 900. (b) Model performance when we vary number of steps with timestamp reward, keeping total steps to be 900. For all the experiments, we report average score of Charades-STA, THUMOS14 and THUMOS15 as in-domain score, and average score of E.T. Bench (Subset) as out-of-domain score.

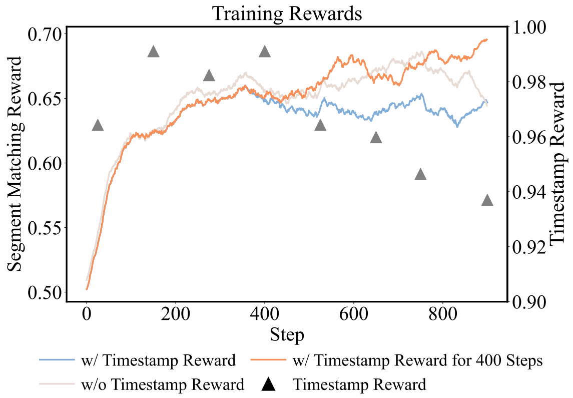


Figure 7: Rewards with different training recipes. We also report timestamp reward during training.

it produces a single segment or at least two segments for a query whose groundtruth consists of more than one segments. As shown in Figure 5 (a), (b), and (c), local matching strategies impose significant penalties on segment matching rewards when model output only contains a single segment, but the penalties imposed by global matching are relatively weak. We further report evolution of numbers of predicted segments during training process in Figure 5 (d). When we remove local matching, numbers of predicted segments significantly drop and their gaps from groundtruths become larger. This indicates that local matching can help better align numbers of predicted segments with groundtruths.

6.2 Design of Phased Rewards

In this section, we explore the effectiveness of our proposed training recipe with phased rewards. We compare it with training model with or without timestamp reward during the whole training process in Table 4. From the table we can see that our proposed recipe of training the model with timestamp reward for 400 steps and without timestamp reward for another 500 steps reaches the highest performance. We further change the total training steps and report the results in Figure 6 (a). We can see that our proposed recipe consistently outperforms other training strategies, showing effectiveness over different data scales.

We also explore model performance when we vary number of steps of keeping timestamp reward. Figure 6 (b) demonstrates that when we train the model with timestamp reward for 400 steps, its performance reaches the peak. To further investigate the reason behind it, we examine values of segment matching reward and timestamp reward during training in Figure 7. Similarly, we observe timestamp reward peaking around 400 steps. If discarding after 400 steps, segment matching reward continues rising, and finally surpassing other training recipes. But if it is kept during the whole training process, segment matching reward would also drop after 400 steps. Removing restriction of referring timestamps in thinking process in the middle of training helps boost model performance.

7 Conclusion

In this work, we introduce MUSEG, a RL-based method to improve video temporal understanding abilities of MLLMs. Experiments demonstrate effectiveness of our method on improving model performance on single-segment and multi-segment grounding tasks, as well as broader time-sensitive scenarios. We hope our proposed method will inspire future research on enhancing temporal understanding abilities of MLLMs.

Limitations

While our method demonstrates strong performance, it is trained exclusively on temporal grounding tasks. We believe that incorporating training data from a wider range of time-sensitive tasks could further improve the performance and generalization capabilities of the trained model. Additionally, although our work primarily focuses on time-sensitive scenarios, we believe that stronger temporal reasoning abilities may also benefit general video understanding tasks by enabling more coherent and structured reasoning. We leave the exploration of how to transfer temporal reasoning capabilities to more general domains as future work.

References

Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lucic, and Cordelia Schmid. 2021. ViViT: A Video Vision Transformer. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6816–6826.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, and 1 others. 2025. Qwen2.5-VL Technical Report. *arXiv preprint arXiv:2502.13923*.

Mu Cai, Reuben Tan, Jianrui Zhang, Bocheng Zou, Kai Zhang, Feng Yao, Fangrui Zhu, Jing Gu, Yiwu Zhong, Yuzhang Shang, and 1 others. 2024. Temporalbench: Benchmarking fine-grained temporal understanding for multimodal video models. *arXiv preprint arXiv:2410.10818*.

Guo Chen, Yicheng Liu, Yifei Huang, Yuping He, Baoqi Pei, Jilan Xu, Yali Wang, Tong Lu, and Limin Wang. 2024. CG-Bench: Clue-grounded Question Answering Benchmark for Long Video Understanding. *arXiv preprint arXiv:2412.12075*.

Yi Chen, Yuying Ge, Rui Wang, Yixiao Ge, Lu Qiu, Ying Shan, and Xihui Liu. 2025. Exploring the Effect of Reinforcement Learning on Video Understanding: Insights from SEED-Bench-R1. *arXiv preprint arXiv:2503.24376*.

Jen-Hao Cheng, Vivian Wang, Huayu Wang, Huapeng Zhou, Yi-Hao Peng, Hou-I Liu, Hsiang-Wei Huang, Kuang-Ming Chen, Cheng-Yen Yang, Wenhao Chai, and 1 others. 2025a. TEMPURA: Temporal Event Masked Prediction and Understanding for Reasoning in Action. *arXiv preprint arXiv:2505.01583*.

Zixu Cheng, Jian Hu, Ziquan Liu, Chenyang Si, Wei Li, and Shaogang Gong. 2025b. V-STaR: Benchmarking Video-LLMs on Video Spatio-Temporal Reasoning. *arXiv preprint arXiv:2503.11495*.

Kaituo Feng, Kaixiong Gong, Bohao Li, Zonghao Guo, Yibing Wang, Tianshuo Peng, Benyou Wang, and Xiangyu Yue. 2025. Video-R1: Reinforcing Video Reasoning in MLLMs. *arXiv preprint arXiv:2503.21776*.

Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. 2017. TALL: Temporal Activity Localization via Language Query. In *Proceedings of the IEEE international conference on computer vision*, pages 5267–5275.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. *arXiv preprint arXiv:2501.12948*.

Yongxin Guo, Jingyu Liu, Mingda Li, Qingbin Liu, Xi Chen, and Xiaoying Tang. 2024. TRACE: Temporal Grounding Video LLM via Causal Event Modeling. *arXiv preprint arXiv:2410.05643*.

Zhenpeng Huang, Xinhao Li, Jiaqi Li, Jing Wang, Xiangyu Zeng, Cheng Liang, Tao Wu, Xi Chen, Liang Li, and Limin Wang. 2024. Online Video Understanding: A Comprehensive Benchmark and Memory-Augmented Method. *arXiv preprint arXiv:2501.00584*.

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*.

H. Idrees, A. R. Zamir, Y. Jiang, A. Gorban, I. Laptev, R. Sukthankar, and M. Shah. 2017. The THUMOS challenge on action recognition for videos "in the wild". *Computer Vision and Image Understanding*, 155:1–23.

Hongyu Li, Jinyu Chen, Ziyu Wei, Shaofei Huang, Tianrui Hui, Jialin Gao, Xiaoming Wei, and Si Liu. 2025a. LLaVA-ST: A Multimodal Large Language Model for Fine-Grained Spatial-Temporal Understanding. *arXiv preprint arXiv:2501.08282*.

Xinhao Li, Ziang Yan, Desen Meng, Lu Dong, Xiangyu Zeng, Yinan He, Yali Wang, Yu Qiao, Yi Wang, and Limin Wang. 2025b. VideoChat-R1: Enhancing Spatio-Temporal Perception via Reinforcement Fine-Tuning. *arXiv preprint arXiv:2504.06958*.

Yun Li, Zhe Liu, Yajing Kong, Guangrui Li, Jiyuan Zhang, Chao Bian, Feng Liu, Lina Yao, and Zhenbang Sun. 2025c. Exploring the Role of Explicit Temporal Modeling in Multimodal Large Language Models for Video Understanding. *arXiv preprint arXiv:2501.16786*.

Ye Liu, Zongyang Ma, Zhongang Qi, Yang Wu, Ying Shan, and Chang W Chen. 2024a. E.T. Bench: Towards Open-Ended Event-Level Video-Language Understanding. *Advances in Neural Information Processing Systems*, 37:32076–32110.

Yuanxin Liu, Shicheng Li, Yi Liu, Yuxiang Wang, Shuhuai Ren, Lei Li, Sishuo Chen, Xu Sun, and Lu Hou. 2024b. Tempcompass: Do video llms really understand videos? *arXiv preprint arXiv:2403.00476*.

Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. 2021. Video Swin Transformer. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3192–3201.

Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. 2021. CLIP4Clip: An Empirical Study of CLIP for End to End Video Clip Retrieval. *Neurocomputing*, 508:293–304.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Viorica Patraucean, Lucas Smaira, Ankush Gupta, Adria Recasens, Larisa Markeeva, Dylan Banarse, Skanda Koppula, Mateusz Malinowski, Yi Yang, Carl Doersch, and 1 others. 2023. Perception Test: A Diagnostic Benchmark for Multimodal Video Models. *Advances in Neural Information Processing Systems*, 36:42748–42761.

Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. 2019. Generalized Intersection Over Union: A Metric and a Loss for Bounding Box Regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal Policy Optimization Algorithms. *arXiv preprint arXiv:1707.06347*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. *arXiv preprint arXiv:2402.03300*.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, and 1 others. 2023. Gemini: A Family of Highly Capable Multimodal Models. *arXiv preprint arXiv:2312.11805*.

Jiawei Wang, Liping Yuan, Yuchen Zhang, and Haomiao Sun. 2024. Tarsier: Recipes for Training and Evaluating Large Video Description Models. *arXiv preprint arXiv:2407.00634*.

Ning Wang, Guangming Zhu, Liang Zhang, Peiyi Shen, Hongsheng Li, and Cong Hua. 2021. Spatio-temporal interaction graph parsing networks for human-object interaction recognition. In *Proceedings of the 29th ACM international conference on multimedia*, pages 4985–4993.

Ye Wang, Boshen Xu, Zihao Yue, Zihan Xiao, Ziheng Wang, Liang Zhang, Dingyi Yang, Wenxuan Wang, and Qin Jin. 2025. TimeZero: Temporal Video Grounding with Reasoning-Guided LVLM. *arXiv preprint arXiv:2503.13377*.

Junbin Xiao, Angela Yao, Yicong Li, and Tat-Seng Chua. 2024. Can I Trust Your Answer? Visually Grounded Video Question Answering. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13204–13214.

Hu Xu, Gargi Ghosh, Po-Yao (Bernie) Huang, Dmytro Okhonko, Armen Aghajanyan, and Florian Metze Luke Zettlemoyer Christoph Feichtenhofer. 2021. VideoCLIP: Contrastive Pre-training for Zero-shot Video-Text Understanding. In *Conference on Empirical Methods in Natural Language Processing*.

Yibin Yan, Jilan Xu, Shangzhe Di, Yikun Liu, Yudi Shi, Qirui Chen, Zeqian Li, Yifei Huang, and Weidi Xie. 2025. Learning Streaming Video Representation via Multitask Training. *arXiv preprint arXiv:2504.20041*.

En Yu, Kangheng Lin, Liang Zhao, Yana Wei, Zining Zhu, Haoran Wei, Jianjian Sun, Zheng Ge, Xiangyu Zhang, Jingyu Wang, and 1 others. 2025. Unhackable Temporal Rewarding for Scalable Video MLLMs. *arXiv preprint arXiv:2502.12081*.

Xingjian Zhang, Siwei Wen, Wenjun Wu, and Lei Huang. 2025. TinyLLaVA-Video-R1: Towards Smaller LMMs for Video Reasoning. *arXiv preprint arXiv:2504.09641*.

A Training Details

We leverage 7B and 3B models of Qwen2.5-VL (Bai et al., 2025) series as our base models. They are trained on large scale image and video data and demonstrate strong instruction following and reasoning abilities. Additionally, there are special designs in Qwen2.5-VL to enable models to process absolute timestamps and dynamic resolutions of video frames. During training and inference of MUSEG-7B and MUSEG-3B, we set maximum total video tokens to be 3584 and maximum number of frames to be 448.

We train MUSEG-7B and MUSEG-3B for 900 steps in total, including 400 steps with timestamp reward and another 500 steps without timestamp reward. We set batch_size $= 14$ and learning_rate $= 1e-5$. We set $\alpha = 2$ in phase 1 and phase 2 reward, and $\beta = 0.4$ in phase 1 reward. Considering that base models have been trained on temporal-related data and already have strong abilities of instruction-following, we do not include SFT stage in our experiments as DeepSeek-R1 (Guo et al., 2025).

B Baselines

We introduce our baselines in Table 2 in this section. We categorize our baselines into SFT-based methods and RL-based methods. We introduce SFT-based models first:

E.T. Chat (~7B): It compresses video frames into single tokens using a Q-Former-based compressor with cross-attention, and generates timestamps with special tokens. It is trained on E.T. Instruct 164k, a dataset covering 9 tasks across 14 sources.

TRACE (~7B): It is trained with a causal event modeling framework, integrating timestamp, salient score, and textual caption prediction tasks. Its training data include 1.9M samples from Valley, TextVR, ShareGPT4Video, and 0.9M samples form ActivityNet Captions and InternVid.

TEMPURA (~3B): It is trained with masked event prediction reasoning, event segmentation and dense captioning tasks. Its training data consist of 500k samples.

Then we introduce RL-based models:

Video-R1 (~7B): It is trained by SFT with 165k samples and RL with 260k samples. Its training data consist of various general image question answering and video question answering tasks.

VideoChat-R1 (~7B): It is trained with temporal grounding, object tracking, video captioning and grounded video question answering tasks, with a total data scale of 18.0k samples.

TimeZero (~7B): It is trained towards temporal grounding tasks. One version of its models is trained with Charades-STA (Gao et al., 2017).