

1	<b>A Limitations</b>	1
2	<b>B Implementation Details</b>	1
3	<b>C Additional Experiments</b>	2
4	<b>D Ablation Studies</b>	3
5	<b>E Qualitative Result</b>	4

## 6 **Important Corrections to Oversights in Main Text**

7 We sincerely apologize for the oversight identified after our main paper submission. These corrections  
8 do not affect the conclusions of the paper. We ensure that our code is fully reproducible, and we  
9 will release the checkpoints, all relevant data, and the benchmark to facilitate verification and further  
10 research by the community. Below, we detail the corrections for the record:

- 11 • **[Correction to RL results in Figure 4 and Table 1]:** In the original submission, the reported RL  
12 results on the Charades and ActivityNet benchmarks were mistakenly taken from the **R1@0.3** scores  
13 instead of the correct **mIoU** values. We have updated the figure in the appendix to reflect the correct  
14 mIoU results. Please refer to Figure S1 for the revised version. Additionally, the Time-R1 result for  
15 R1@0.3 on ActivityNet in Table 1 should be 58.6, not 58.1.
- 16 • **[Statement removal of “VideoQA” in line 182]:** We initially considered incorporating QA data  
17 to preserve performance on the QA benchmarks. However, we found that training solely on the TVG  
18 task does not harm the model’s performance on the QA tasks. As a result, the final model was trained  
19 solely on TVG data. The mention of using QA data was a clerical oversight.

## 20 **A Limitations**

21 Despite achieving notable improvements on the TVG task, our approach still has several limitations.  
22 First, Time-R1 suffers from slower training and inference speeds, primarily due to its large model  
23 size and reliance on autoregressive text generation. Second, to manage GPU memory consumption,  
24 we use a relatively low frame sampling rate, which may result in the loss of fine-grained motion  
25 information across frames. Finally, Time-R1 currently cannot handle ultra-long videos, limiting its  
26 applicability in scenarios such as full-length movie understanding.

## 27 **B Implementation Details**

28 **Details of Time-R1 framework.** Inspired by DAPO [11], we adopt its token-level loss for training,  
29 rather than the sample-level loss used in GRPO. Apart from minor changes to the loss, all setting  
30 is identical to GRPO. Besides, we find that other techniques introduced in DAPO do not benefit  
31 the TVG task, thus aborting other techniques. We update the model parameters at every step, thus  
32  $\frac{\pi_{\theta}(o_i)}{\pi_{\theta_{\text{old}}}(o_i)} = 1$ . The sample number  $G$  is set to 8. The coefficient  $\beta$  is set to 0.04.

33 **Details of TimeRFT training.** For RFT data filtering, we use a Gaussian distribution with a fixed  
34 variance of 0.2, while varying the mean to control sample selection. In our cold start phase, we  
35 construct 150 samples from our training data sources (e.g., YT-Temporal [10]) to fine-tune the LLM  
36 using LoRA [4], with a LoRA rank of 64 and a LoRA alpha of 128. All of our results are reported  
37 based on the final training epoch. For RL, we use a learning rate of 1e-6 with the AdamW optimizer  
38 with  $\beta_1=0.9$ ,  $\beta_2=0.999$ , and a linear scheduler to decay the learning rate from 1e-6 to 0. We use a  
39 batch size of 8 with gradient accumulation set to 2.

40 **Details of our evaluation prompts.** As shown in Figure S10, for Temporal Video Grounding, the  
41 prompts used for training and testing are designed to encourage the model to reason before responding,  
42 following a template-based answer format. For VideoQA, we have two versions of prompts: one with  
43 Chain-of-Thought (CoT) and one without CoT.

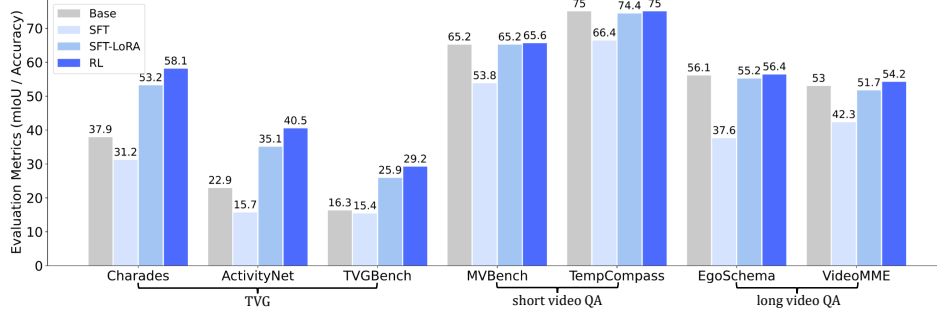


Figure S1: Comparison between post-training paradigms across various tasks, including short video QA, long video QA and temporal video grounding. We use mIoU as metric.

Table S1: Comparison of different approaches on TVGBench for all types. We use mIoU as metric.

Method	EC	ES	HAC	HAP	HAS	HP	OA	OC	OEC	OES	OT
TimeChat [7]	22.3	32.8	16.6	9.8	14.6	35.1	15.0	9.2	2.4	18.0	10.2
TimeSuite [12]	27.3	39.6	14.2	12.8	24.9	39.6	14.6	13.9	6.7	32.6	14.3
TRACE [3]	57.1	66.8	25.9	17.5	26.5	45.1	17.8	22.1	12.5	36.8	24.9
VideoChat-Flash [6]	38.3	47.2	12.9	13.9	27.1	39.4	14.9	12.7	6.5	24.3	12.9
Gemini-2.5-Pro [2]	46.7	45.3	21.1	27.6	30.9	39.9	23.0	31.1	14.1	35.9	17.8
Time-R1 (ours)	49.3	65.3	28.3	24.3	39.3	56.2	26.3	21.8	9.0	32.7	21.8

**Details of TVG baseline methods and implementations.** We evaluate the baselines on TVGBench using their original best-performing setting, focusing primarily on video input and prompt design.

- TimeChat [7] is built upon the InstructBLIP [1] architecture and introduces a video Q-former to encode video tokens. It operates at a resolution of 224 and samples 96 frames.
- TRACE [3] treats each combination of timestamp, saliency score, and caption as a discrete event and enables the LLM to autoregressively generate event sequences. It operates at a higher resolution of 336 and samples 128 frames.
- TimeSuite [12] introduces a token shuffling strategy to compress long video token sequences and incorporates positional encoding to enhance visual understanding. It adopts a resolution of 224 and samples 128 frames.
- VideoChat-Flash [6] proposes a progressive visual token dropping mechanism within intermediate LLM layers to compress video inputs and extend the effective context length. It uses a resolution of 448 and samples video at 1 fps, with a maximum of 512 frames.
- Gemini-2.5-Pro [2]: Gemini-2.5-Pro is state-of-the-art video understanding model capable of reasoning over videos exceeding one hour in length. It supports video question answering and temporal localization tasks.

**Details of our implemented SFT baselines.** We implemented two versions of SFT fine-tuning: one is full-parameter fine-tuning of the LLM (SFT), and the other is LoRA-based fine-tuning of the LLM (SFT-LoRA). For SFT-LoRA, the LoRA rank is set to 64, and the LoRA alpha is set to 128. Both configurations use the following settings: a learning rate of  $2e-5$ , the AdamW optimizer with  $\beta_1=0.9$ ,  $\beta_2=0.999$ , a weight decay of 0, the batch size of 8, and accumulation steps of 2. We fine-tune for 5 epochs on our 2.5K data, and use a linear scheduler to gradually decay the learning rate to 0.

## C Additional Experiments

**Generalization of RL vs. SFT.** For SFT, LoRA proves to be a more effective setting for activating the model’s capabilities. Therefore, we further implement a LoRA-based version of SFT (SFT-LoRA) for comparison with RL. As shown in Figure S1, SFT-LoRA can also significantly boost the model’s performance. For example, SFT-LoRA improves performance from 37.9 to 53.2 on Charades, but it still lags behind RL, which reaches 58.1, suggesting that RL is more data-efficient. Moreover, on QA benchmarks, although SFT-LoRA better preserves the model’s original capabilities than SFT, performance degradation remains. For instance, on VideoMME, SFT-LoRA drops from 53 to 51.7, whereas RL improves the performance to 54.2.

Table S2: Ablation of RFT data filtering strategies. We use mIoU as metric.

Method	R1@0.3	R1@0.5	R1@0.7	mIoU
random	39.4	26.5	16.4	27.4
gaussian (0.3)	41.6	28.5	15.6	28.6
gaussian (0.5)	40.6	28.2	16.0	28.3
gaussian (0.7)	37.2	26.9	15.5	26.5
uniform	40.4	28.5	15.9	28.3

Table S3: Ablation of KL and thinking process in GRPO. We use mIoU as metric.

KL	CoT	R1@0.3	R1@0.5	R1@0.7	mIoU
✗	✗	40.4	29.1	14.9	28.1
✓	✗	40.8	27.4	15.0	27.7
✗	✓	42.9	29.5	15.0	29.1
✓	✓	41.6	28.5	15.6	28.6

Table S4: Comparison of the token-level loss design used by DAPO [11] and the sample-level loss design used by GRPO [8].

loss	Charades-STA				ActivityNet				TVGBench			
	R1@0.3	R1@0.5	R1@0.7	mIoU	R1@0.3	R1@0.5	R1@0.7	mIoU	R1@0.3	R1@0.5	R1@0.7	mIoU
GRPO	76.7	59.8	34.4	57.0	55.9	37.1	20.3	37.8	40.8	28.0	16.5	28.4
DAPO	77.4	60.0	34.1	57.2	56.2	37.4	20.4	38.0	41.6	28.5	15.6	28.6

**In-depth comparisons of different approaches on TVGBench by semantic type.** As shown in Table S1, the table provides a detailed performance comparison of various methods on the TVGBench dataset across different semantic categories. Specifically, the abbreviations represent: EC (Environment Change), ES (Environment State), HAC (Human Action – Complex), HAP (Human Action – Procedural), HAS (Human Action – Simple), HP (Human Pose), OA (Object Attribute), OC (Object Counting), OEC (Object Existence – Complex), OES (Object Existence – Simple), and OT (Object Transition). Detailed definition and construction process can be found in Figure S11.

Time-R1 demonstrates strong competitiveness across multiple semantic categories. Particularly in the four tasks of HAC, HAS, HP and OA, Time-R1 achieved the highest scores among all compared methods, showcasing its excellent ability in understanding the details of human actions and identifying object features. For instance, it reached 56.2 on HP and 26.3 on OA. In the three tasks of ES, EC and OT, Time-R1 demonstrates strong performance comparable to the top model TRACE, with its performance being very close or immediately following. In the HAP task, Time-R1 also performs excellently, with its performance being in the same tier as Gemini-2.5-Pro. In contrast, tasks such as OC, OEC and OES pose universal challenges for all existing video understanding models. Especially for the OEC task, the performance of all models shows significant room for improvement, reflecting the inherent high difficulty of such tasks.

**Comparison of speed and accuracy between inference library transformers and vllm.** We observe that the inference speed of the implementation in the transformers [9] library is very slow. To address this, we implemented an accelerated inference version using vLLM [5] for all downstream benchmarks. For example, on TVGBench, the vLLM-based implementation required only 502 seconds to infer 800 samples using 8 GPUs, whereas the transformers library implementation took 2520 seconds. This achieves an overall speedup of  $5\times$ .

## D Ablation Studies

**Ablation of different RFT data filtering strategies.** As shown in Table S2, different data filtering strategy in the initial round affects model’s performance. First, appropriate Gaussian filtering outperforms both uniform and random filtering methods. Among the Gaussian filtering settings, a standard deviation of 0.3 yields the best results, followed by 0.5 and then 0.7. These findings suggest that incorporating moderately challenging samples during RFT helps improve the model’s generalization capability more effectively than using either overly easy or extremely difficult examples.

**Ablation of KL and CoT during GRPO training.** As shown in Table S3, incorporating CoT reasoning during training leads to improved performance compared to the No-CoT setting, suggesting that CoT enhances the model’s temporal video grounding capabilities. When KL divergence is omitted (No-KL), performance slightly decreases under the No-CoT setting but unexpectedly improves when CoT is present. However, we find that in the No-KL+CoT setting, the model often fails to produce thinking process, directly jumping to answers. In contrast, using KL divergence helps maintain more logical reasoning that is easier to follow. To balance performance and interpretability, we adopt a training setup that includes both KL and CoT.

Table S5: Performance comparison of different model sizes.

Method	Charades-STA				ActivityNet				TVGBench			
	R1@0.3	R1@0.5	R1@0.7	mIoU	R1@0.3	R1@0.5	R1@0.7	mIoU	R1@0.3	R1@0.5	R1@0.7	mIoU
TimeZero-3B	74.6	53.1	26.0	51.2	40.0	21.0	8.7	23.2	33.5	21.0	10.5	21.7
TimeZero-3B*	78.7	64.1	36.9	59.9	66.8	46.8	24.7	46.1	-	-	-	-
TimeZero-7B	78.1	60.8	35.5	58.1	58.1	39.0	21.4	40.5	41.8	29.4	16.4	29.2
TimeZero-7B*	82.8	72.2	50.1	60.9	73.3	55.6	34.0	52.1	-	-	-	-

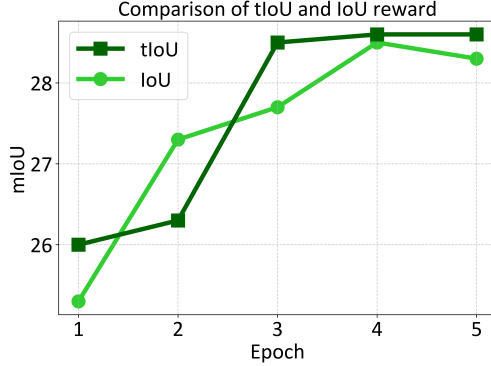


Figure S2: Performance comparison of tIoU and IoU in multi-epoch training.

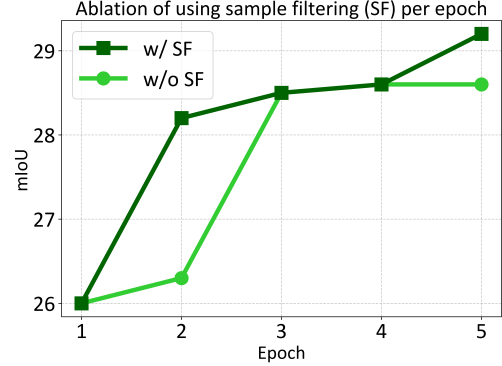


Figure S3: Ablation of data filtering in multi-epoch training.

**Comparison of tIoU and IoU during multi-epoch training.** As shown in Figure S2, tIoU consistently outperforms standard IoU during both the early and late stages of training over the first 5 epochs. Notably, while tIoU steadily improves as training progresses, IoU shows a decline in performance by the fifth epoch. This highlights the advantage of using tIoU as a more stable and reliable evaluation metric for temporal grounding.

**Ablation of data filtering in multi-epoch training.** As shown in Figure S3, applying sample filtering (SF) to remove simpler training samples yields consistent performance improvements across epochs. This suggests that easy samples may introduce noise or reduce the effectiveness of learning, and filtering them helps focus the model on more informative and challenging instances.

**Ablation of DAPO & GRPO.** The sample-level loss used by GRPO computes the loss by averaging over each individual sample. This approach leads to unequal loss contributions for tokens when dealing with CoTs of varying lengths. DAPO addresses this issue by employing a token-level loss. The underlying principle is that the token-level loss can effectively guide the model in the process of CoT generation, allowing it to learn useful patterns from CoTs of different lengths sampled during training. In Table S4, we compare these two loss designs. We find that DAPO outperforms GRPO on the majority of metrics, thus we adopt DAPO’s loss design.

**Different Model Size.** Table S5 presents a performance comparison of different model sizes. These results indicate that larger models achieve better zero-shot performance and continue to outperform smaller models after fine-tuning. These findings support the notion that scaling up model capacity enhances generalization and leads to superior results on the TVG tasks.

## E Qualitative Result

**Case study of temporal video grounding on Charades and ActivityNet.** As shown in Figure S4, in the example above, given a relatively complex language instruction, Time-R1 demonstrates more accurate localization than all baselines, successfully capturing the initial event “goes back to the pink bucket” within the timestamp, while other SoTA models like Gemini-2.5-Pro fail. In the example below, the model accurately localizes the event, excluding “a person is lying on the couch” and correctly distinguishing between sitting and lying, unlike other models which either localize only a small segment (TimeSuite and VideoChat-Flash) or the entire segment (TRACE and Gemini-2.5-Pro).

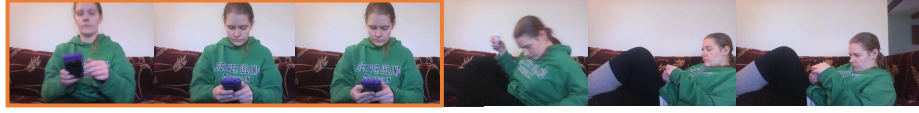


**Question:** Where is the event "The man goes back to the pink bucket to wash another dark item clothing, he wrings it out in the pink bucket, throws it in the yellow bucket, then wrings his hands together a few times while staring at the camera"?

(video length: 74s)

Ground Truth	33.1s	73.5s
TimeSuite	20.0s	30.0s
Gemini-2.5-Pro	57.0s	73.0s
VideoChat-Flash	0.0s	17.2s
Trace	54.9s	74.1s
Time-R1 (ours)	35.0s	75.0s

**Time-R1 thinking (ours):** The event "The man goes back to the pink bucket to wash another dark item clothing, he wrings it out in the pink bucket, throws it in the yellow bucket, then wrings his hands together a few times while staring at the camera." occurs between 35.00 and 75.00 seconds in the video.



**Question:** Where is the event "a person is seen sitting on a couch"?

(video length: 31s)

Ground Truth	0.0s	12.8s
TimeSuite	0.0s	5.0s
Gemini-2.5-Pro	0.0s	30.0s
VideoChat-Flash	0.0s	5.0s
Trace	0.0s	31.6s
Time-R1 (ours)	0.0s	10.0s

**Time-R1 thinking (ours):** The event "a person is seen sitting on a couch" occurs at the beginning of the video. The person is seated on a couch, holding a phone, and appears to be engaged with it. This aligns with the initial frames of the video.

Figure S4: Temporal grounding cases from Charades and ActivityNet benchmark. **Time-R1** outperforms other SoTA models.

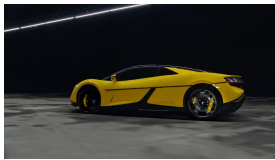


(video length: 38m) **Question:** What do heroes of legend use to defeat the enemy based on the video?

- (A) Their wisdom  
(C) Their superpower

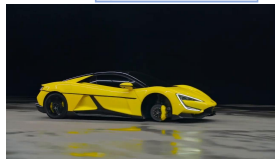


- (B) A big robot  
(D) Power of music



(video length: 17s) **Question:** What's wrong with this car?

- (A) It doesn't have a left rear wheel.  
(C) Its headlamp is broken.



- (B) It doesn't have a right front wheel.  
(D) Its right door is broken.



Figure S5: Case study on VideoMME (w/o CoT), demonstrating that **Time-R1** achieves better performance than the base model.



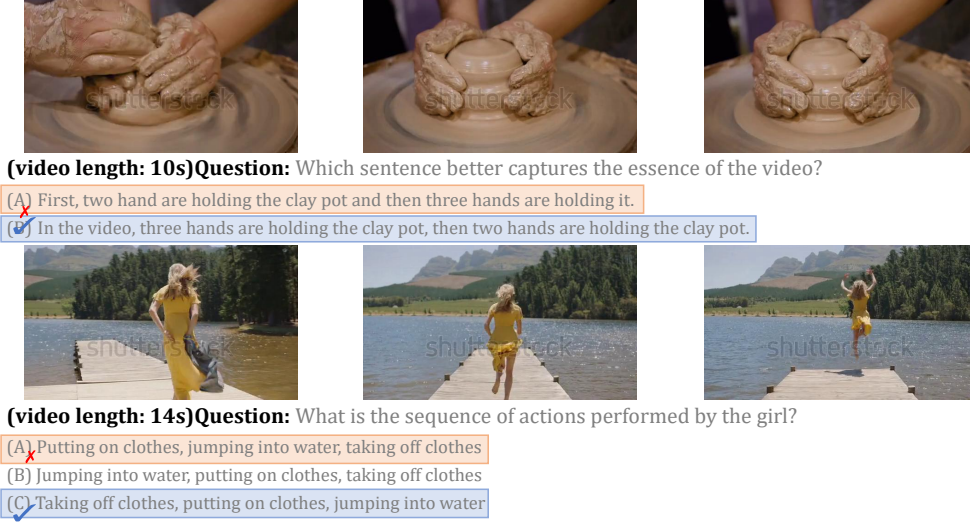


Figure S6: Case study on TempCompass (w/o CoT), demonstrating that **Time-R1** achieves better performance than the **base** model.

141 **Case study of short video QA on VideoMME and TempCompass.** As shown in Figures S5 and S6,  
 142 Time-R1 demonstrates improved performance over the base model in tasks requiring positional  
 143 judgment, scene storyline reasoning, and visual reasoning. For example, in Figure S5, Time-R1  
 144 correctly identifies that a car in the video is missing its right-front wheel, a detail that the base model  
 145 fails to recognize. This reflects that Time-R1 likely possesses stronger video localization capabilities,  
 146 which in turn enhance its visual reasoning ability. In Figure S7, we output a CoT when answering  
 147 the QA task, providing some interpretability. This example shows that Time-R1’s reasoning process  
 148 is more concise, whereas the base model often reasons correctly but arrives at the wrong answer.  
 149 This suggests that Time-R1’s reasoning may be more effective in guiding the final answer, possibly  
 150 benefiting from the outcome-driven RL of GRPO.

151 **Case study of long video QA on EgoSchema and VideoMME.** Figure S8 presents a long egocentric  
 152 video QA example focused on summarizing task steps. In the "Hanging the Dress" case, the base  
 153 model fails to identify all key steps, while our Time-R1 model correctly selects the answer by  
 154 generating a more accurate chain-of-thought (CoT). In Figure S9, the task involves identifying the  
 155 primary tools and ingredients used in the video and understanding their respective contributions.  
 156 An additional example in Figure S5 involves animated scene reasoning, where Time-R1 correctly  
 157 infers—based on elements like the presence of a guitar—that the heroes defeated the enemies using  
 158 the power of music, whereas the base model misinterprets the cause. Across these complex reasoning  
 159 tasks involving long videos, Time-R1 consistently demonstrates superior performance compared to  
 160 the base model.

161 **Illustration of our prompt at training and inference time.** Figure S10 presents the prompts used  
 162 for the temporal video grounding and video QA tasks at both training and inference time.

163 **Illustration of our prompt to annotate query semantics on TVGBench.** Figure S11 presents the  
 164 prompt used to annotate query semantics in the TVGBench dataset. The prompt is designed to guide  
 165 the DeepSeek LLM in classifying each input query into one of 11 predefined semantic categories.  
 166 To improve annotation quality, we refer to the construction of existing benchmarks and carefully  
 167 select both positive and negative examples for each semantic type. These examples are chosen to  
 168 include queries that are either easy or difficult for the model to answer, helping to refine the model’s  
 169 understanding and improve labeling accuracy.



(video length: 8s) Question: Is the pineapple being pushed forward by a person?

(A) Yes



(B) No



#### Base model thinking:

To determine if the pineapple is being pushed forward by a person, let's analyze the video:

1. The video shows a pineapple on a pink surface.
2. A hand appears in the frame and interacts with the pineapple.
3. The hand lifts the pineapple, indicating that it is being moved by the person.

Given this analysis, the pineapple is indeed being pushed forward by a person.

#### Time-R1 thinking (ours) :

The video shows a hand reaching into the frame and grasping the pineapple. The hand then lifts the pineapple, indicating that the pineapple is not being pushed forward by a person but rather being moved by the hand.



(video length: 9s) Question: In which sequence does the man perform the actions?

(A) Gliding on the skateboard and then jumping

(B) Jumping and then tying his shoelaces

(C) Tying his shoelaces and then gliding on the skateboard

(D) Gliding on the skateboard and then tying his shoelaces



#### Base model thinking:

Let's analyze the sequence of actions performed by the man in the video:

1. The man is initially seen standing on a skateboard.
2. He then proceeds to tie his shoelaces.
3. After tying his shoelaces, he glides on the skateboard.

Given this sequence, the correct answer is: (D) gliding on the skateboard and then tying his shoelaces

#### Time-R1 thinking (ours) :

The sequence of actions performed by the man in the video is as follows: first, he ties his shoelaces, and then he glides on the skateboard. This can be observed from the video where the man is seen tying his shoelaces before he starts gliding on the skateboard.

Figure S7: Case study on TempCompass (w/ CoT), demonstrating that Time-R1 achieves better performance than the base model.



**(video length: 3m) Question:** Summarize the steps taken by c to complete the tasks in this video. don't list every action narrated, but rather focus on major components and how they're executed.

(A) C folds the dress, places it on the ironing board, and then hangs it up.

(B) C folds the dress, places it on the ironing board, and then packs it.

(C) C folds the dress, places it on the ironing board, and then washes it.

☒ (D) C folds the dress, places it on the ironing board, and then irons it and hangs it up.

☐ (E) C folds the dress, places it on the ironing board, irons it, and then folds it again.

### Base model thinking:

Let's analyze the video step by step to determine the correct sequence of actions:

1. Folding the Dress: The video starts with C folding a blue dress on the ironing board. This is the first major action.
2. Placing the Dress on the Ironing Board: After folding the dress, C places it on the ironing board.
3. Ironing the Dress: C then irons the dress, ensuring it is smooth and free of wrinkles.
4. **Hanging the Dress:** Finally, C hangs the dress up, completing the task.

Now, let's evaluate the options:

(A) C folds the dress, places it on the ironing board, and then hangs it up.

(B) C folds the dress, places it on the ironing board, and then packs it.

(C) C folds the dress, places it on the ironing board, and then washes it.

(D) C folds the dress, places it on the ironing board, and then irons it and hangs it up.

(E) C folds the dress, places it on the ironing board, irons it, and then folds it again.

The correct sequence of actions matches option (D).

### Time-R1 thinking (ours) :

The video shows C folding a dress, placing it on the ironing board, and then ironing it. The sequence of actions is clear and follows a logical order. The dress is folded, placed on the ironing board, and then ironed, which is a common sequence for completing a task like ironing a dress.

Figure S8: Case study on EgoSchema (w/ CoT), demonstrating that Time-R1 achieves better performance than the base model.





**(video length: 3m) Question:** Identify the primary tools and ingredients c used throughout the video, and discuss how they contributed to the overall dessert preparation process.

(A) The primary tools and ingredients c used throughout the video were a spatula, a strainer, a dessert bowl, cocoa powder, and tissue paper. the spatula was used to spread the batter in the bowl. the strainer was used to sieve the cocoa powder into the bowl. the dessert bowl was used to hold the batter. the cocoa powder was used to add flavor and color to the batter. the tissue paper was used to wipe the edges of the bowl.

(B) The primary tools and ingredients c used throughout the video were a mixing bowl, a whisk, a baking dish, flour, sugar, eggs, and butter. the mixing bowl was used to mix the batter. the whisk was used to beat the eggs. the baking dish was used to bake the cake. the flour was used to add structure to the cake. the sugar was used to add sweetness to the cake. the eggs were used to add moisture to the cake. the butter was used to add flavor and richness to the cake.

(C) The primary tools and ingredients c used throughout the video were a muffin tin, a spoon, chocolate chips, flour, sugar, eggs, and butter. the muffin tin was used to bake the muffins. the spoon was used to scoop the batter into the muffin tin. the chocolate chips were used to add flavor and color to the muffins. the flour was used to add structure to the muffins. the sugar was used to add sweetness to the muffins. the eggs were used to add moisture to the muffins. the butter was used to add flavor and richness to the muffins.

(D) The primary tools and ingredients c used throughout the video were a cake pan, a spatula, chocolate frosting, flour, sugar, eggs, and butter. the cake pan was used to bake the cake. the spatula was used to spread the frosting on the cake. the chocolate frosting was used to add flavor and color to the cake. the flour was used to add structure to the cake. the sugar was used to add sweetness to the cake. the eggs were used to add moisture to the cake. the butter was used to add flavor and richness to the cake.

(E) The primary tools and ingredients c used throughout the video were a mixing bowl, a whisk, a baking sheet, flour, sugar, eggs, and butter. the mixing bowl was used to mix the batter. the whisk was used to beat the eggs. the baking sheet was used to bake the cookies. the flour was used to add structure to the cookies. the sugar was used to add sweetness to the cookies. the eggs were used to add moisture to the cookies. the butter was used to add flavor and richness to the cookies.

Figure S9: Case study on EgoSchema (w/o CoT), demonstrating that Time-R1 achieves better performance than the base model.

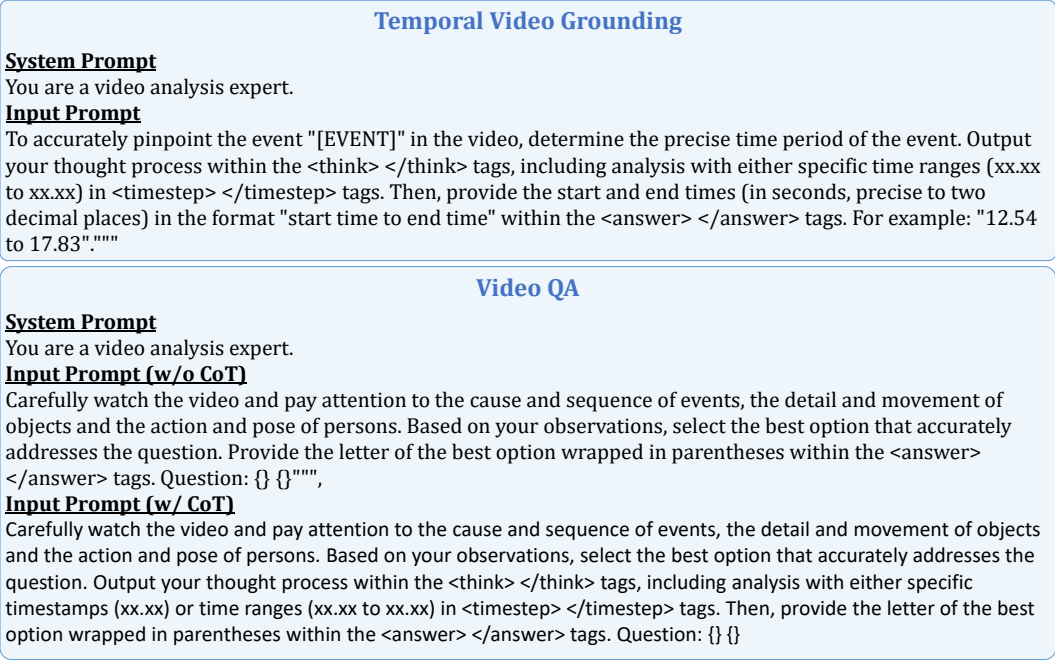


Figure S10: Illustration of prompts at both training and inference time.

## ## Task

Classify input queries into exactly one of the following categories based on their semantic content:

### 1. Human Action (Simple)

- **Definition:** Singular physical movements or basic interactions.
- **Examples:** - person opens a book over their head. - The person gets out some ginger. - who did I talk to in the shopping mall?

### 2. Human Action (Complex)

- **Definition:** Single continuous event with intricate components or concurrent elements.
- **Examples:** - He is talking while several people are using rowing machines.  
- One man wearing blue shirt wearing a jumping leg extension and another man wearing red pants play on a field.  
- who did I interact with when I did activity of fixing camping tent?

### 3. Human Action (procedural)

- **Definition:** contains multiple sequential events with explicit temporal boundaries. contains multiple actions, each with a clear start and end.
- **Examples:** - The person procures a condiment from the pantry, takes a spoon from the drawer which he uses to scoop it into the pan, then returns the condiment to the pantry, places the spoon in the sink and again stirs the pan.  
- The person takes out a spoon from the drawer, scoops some sugar into the glass, stirs it with the juice, and returns the package to the pantry.
- **Negative Examples:** - Then the man juices some lemons in a juicer: only one action  
- She gets out a cutting board and knife: only one action  
- He then finishes by doing tricks: only one action  
- She removes bits of shell until there is a small hole: only one action

### 4. Human Pose

- **Definition:** Static body positions or group configurations. Posture descriptors, positional prepositions
- **Examples:** - Several other people are in the background working out on the equipment.  
- A young child is seen standing before a set of monkey bars.

### 5. Object Existence (Simple)

- **Definition:** Current location/status queries. Simple location prepositions.
- **Examples:** - Where is the tap?  
- where is the chopsticks?  
- In what location did i see the blue tent?

### 6. Object Existence (Complex)

- **Definition:** Queries about historical object positions changed by human actions, requiring temporal-action context (e.g., "after/before [action]").
- **Examples:** - Where was the spatula after I first used it?  
- Where was the sieve before I picked it?  
- what bolt did I pick?  
- What mushroom did i chop

### 7. Object Attribute

- **Definition:** Physical/abstract property inquiries. Property descriptors (color/size/material)
- **Examples:** - what material did I pick from the shelf?  
- what color is the toilet bin?

### 8. Object Counting

- **Definition:** Quantitative object presence queries. Numeric quantifiers, plural objects
- **Examples:** - how many tissue paper were on the floor?  
- how many rolls are in the tray

### 9. Object Transition

- **Definition:** State/position change confirmation. Transformation verbs, completion checks
- **Examples:** - The bulb is broken apart.  
- Did I close fridge?,

### 10. Environment Change

- **Definition:** Dynamic scene modifications. Transient elements, overlay content
- **Examples:** - video ends with clothes/captions scrolling down

### 11. Environment State

- **Definition:** Persistent scene elements. Static overlays, permanent fixtures
- **Examples:** - Intro states 'Progression: Lisa's First Season'  
- 'Trend Routing Technology' logo appears

## ## Output Format

Return ONLY the exact category name from:[Human Action (Procedural), Human Action (Complex), Human Action (Simple), Human Pose, Object Existence (Simple), Object Existence (Complex),Object Attribute, Object Counting, Object Transition, Environment Change, Environment State]"

INPUT\_PROMPT = ""Given the query below, classify it into one of the categories mentioned above.Query: {query} Your response:

Figure S11: Prompts for LLM used to annotate the semantics of each query on TVGBench

## References

- [1] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning, 2023. [2](#)
- [2] Google DeepMind. Gemini 2.5: Our most intelligent ai model. *Google DeepMind*, 2025. Model ID: gemini-2.5-pro-preview-03-25. [2](#)
- [3] Yongxin Guo, Jingyu Liu, Mingda Li, Qingbin Liu, Xi Chen, and Xiaoying Tang. Trace: Temporal grounding video llm via causal event modeling. *arXiv preprint arXiv:2410.05643*, 2024. [2](#)
- [4] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022. [1](#)
- [5] Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023. [3](#)
- [6] Xinhao Li, Yi Wang, Jiashuo Yu, Xiangyu Zeng, Yuhao Zhu, Haian Huang, Jianfei Gao, Kunchang Li, Yanan He, Chenting Wang, Yu Qiao, Yali Wang, and Limin Wang. Videochat-flash: Hierarchical compression for long-context video modeling. *arXiv preprint arXiv:2501.00574*, 2024. [2](#)
- [7] Shuhuai Ren, Linli Yao, Shicheng Li, Xu Sun, and Lu Hou. 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, 2024. [2](#)
- [8] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. [3](#)
- [9] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, October 2020. Association for Computational Linguistics. [3](#)
- [10] Antoine Yang, Arsha Nagrai, Paul Hongsuck Seo, Antoine Miech, Jordi Pont-Tuset, Ivan Laptev, Josef Sivic, and Cordelia Schmid. Vid2seq: Large-scale pretraining of a visual language model for dense video captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10714–10726, 2023. [1](#)
- [11] Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025. [1](#), [3](#)
- [12] Xiangyu Zeng, Kunchang Li, Chenting Wang, Xinhao Li, Tianxiang Jiang, Ziang Yan, Songze Li, Yansong Shi, Zhengrong Yue, Yi Wang, Yali Wang, Yu Qiao, and Limin Wang. Timesuite: Improving MLLMs for long video understanding via grounded tuning. In *The Thirteenth International Conference on Learning Representations*, 2025. [2](#)