## On the Consistency of Video Large Language Models in Temporal Comprehension

# Supplementary Material

In this file, we provide additional details as follows:

- A. Details of Evaluation Datasets: We provide the details of our proposed datasets, Charades-CON and ActivityNet-CON.
- B. **Details of Models**: We describe the Video-LLMs that are utilized in our experiments one by one. Then we provide the experiment setup of the models for our evaluation. Additionally, we further discuss the limitations of the time-aware models.
- C. **Details of Prompting Methods**: We describe two prompting methods, including chain-of-thought and description prompting.
- D. **Details of Instruction Tuning**: We detail the process of instruction tuning and VTune.
- E. Additional Experiments: We provide additional experiments to provide further insights.
- F. **Prompt Designs**: In this study, we design various prompts to construct evaluation sets, evaluate the model's responses, and generate verification queries. We provide detailed prompts for each.

## A. Details of Datasets

In Table 5, we provide statistics of each dataset. The length of the video and their moments remain almost the same compared to the original datasets. Generated aligned and misaligned queries have similar lengths compared to the ground-truth queries.

## **B.** Details of Video-LLMs

#### **B.1. Video-LLMs**

In this section, we describe eight open-source Video-LLMs: Video-LLaMA [41], Video-LLaVA [20], Video-ChatGPT [24], VideoChat2 [17], Video-LLaMA2 [4], TimeChat [31], VTimeLLM [9], and VTG-LLM [8], which are utilized in our evaluation. Note that we experiment with a single A100 GPU for all models.

1. Video-LLaMA exhibits two branches: Vision-Language and Audio-Language in its modeling and utilizes crossmodal training from both the frozen pre-trained visual and audio encoder. It shows a remarkable zero-shot audio understanding capability and also generates responses to the visual and audio information presented in the videos. We select the fine-tuned checkpoint: *Video-LLaMA-2-7B-Finetuned*, which is additionally fine-tuned on the instruction tuning data from Mini-GPT-4 [46], LLaVA [21], and VideoChat [16].



Figure 7. Consistency evaluation results of Video-ChatGPT and VideoChat2 using different prompting methods. While the prompting methods improve Video-ChatGPT, they are ineffective for VideoChat2.

Dataset	Duration	Moment	GT Query	Aligned Query	Misaligned Query
Charades-CON	30.7s	8.1s	7.5	10.6	8.3
ActivityNet-CON	132.4s	39.6s	12.2	12.4	11.3

Table 5. Statistics of each dataset. We provide the average number of each column. The numbers for the queries represent the average word count.

- Video-LLaVA enables an LLM to perform visual reasoning capabilities, and it leads to comprehending both images and videos simultaneously. To do this, Video-LLaVA collects both image and video-text datasets and incorporates them in its instruction tuning.
- 3. Video-ChatGPT designs spatiotemporal video modeling and constructs video instruction tuning upon LLaVA [21]. It introduces a new dataset for video instruction tuning, containing 100,000 high-quality video-instruction pairs. Video-ChatGPT outperforms previous Video-LLMs in Zero-shot VQA across several benchmarks. Additionally, Video-ChatGPT proposes a video conversation evaluation framework.
- 4. **VideoChat2** design a progressive multimodal training. Specifically, they aim to align vision and text in the first stage and then generate captions from various image-text pairs. Finally, they conduct instruction tuning to better

Expected Answer	Templates
Yes	Is the event $q$ present from $m$ in the video? Is the event $q$ occurring from $m$ in the video? Does the event $q$ happen from $m$ in the video? Is the event $q$ included from $m$ in the video?
No	Is the event $q$ absent from $m$ in the video? Is the event $q$ not present from $m$ in the video? Does the event $q$ not happen from $m$ in the video? Is the event $q$ missing from $m$ in the video?

Table 6. Templates for holistic verification. Given the query sentence q and the model's predicted timestamp m, the above question templates can be used for holistic verification.

align responses with human instructions. VideoChat2 demonstrates significant improvements in Zero-Shot Video Question Answering, video conversation, and own built comprehensive video benchmarks.

- 5. Video-LLaMA2 is one of the state-of-the-art Video-LLMs, demonstrating superior performances on video question-answering tasks. They seamlessly integrate both visual and audio modalities in videos and propose STC connector to understand spatiotemporal video information.
- 6. TimeChat is specifically developed to accurately localize and understand specific video moments from long videos. TimeChat designs two key architectural: (1) A time-aware frame encoder that explicitly encodes video frames along with timestamps, (2) A sliding video Q-Former to accommodate sequential information in video frames. Specifically, TimeChat borrows the image Q-Former from InstructBLIP [5] and the video Q-former from Video-LLaMA [41]. Additionally, TimeChat includes the instruction tuning datasets of Valley [23] and constructs 125K video instruction tuning datasets to perform time-related video understanding tasks, such as VTG and DVC.
- 7. **VTimeLLM** proposes a three-stage temporal-aware method, including image-text training and understanding events within the video, enabling more precise video temporal understanding. VTimeLLM devises two types of QA dialogue templates, including single-turn and multi-turn, to prompt questions requiring a comprehensive description of all events and their corresponding times-tamps.
- VTG-LLM utilizes specific modules to improve temporal representations in Video-LLM. While previous Video-LLMs suffer quantization errors by using special time tokens, VTG-LLM use absolute-time tokens (*e.g.*, <TIME\_ZERO>, <TIME\_ONE>) to resolve this and handle videos more than one hour. The model is instruction-tuned with VTG-IT-120K, a high-quality and comprehensive instruction tuning for time-related video understand-

Method	Template
[1]	When does the event $q$ happen in the video? Please only return its start time and end time.
[2]	Please find the visual contents in the video described by a given event, determining its starting. The format should be: 'start time - end seconds'. Now I will give you the event: q. Please only return its start time and end time.
[3]	Please answer when the event $q$ occurs in the video. The output format should be: 'start - end seconds'. Please return its start time and end time."
TimeChat	Please find the visual event described by a sentence in the video, determining its starting and ending times. The format should be: 'The event happens in the 'start time - end time'. For example, The event 'person turn a light on' happens in the 24.3 - 30.4 seconds. Now I will give you the textual sentence: $q$ . Please return its start time and end time.
VTimeLLM	During which frames can we see the event $q$ ?
VTG-LLM	Find the video segment that corresponds to the given textual query $q$ and determine its start and end seconds.
GPT-40	In which time frames does the event $q$ happen? Iden- tify the frames where the event takes place and provide the answer like 'start - end seconds.'. For example, if you think the given event occurs from 5 to 10 seconds in the video, please return the 'The given happens in 5 - 10 seconds.' Please only include the timestamp in the answer without any other content.
Gemini 1.5 Flash	When does the event $q$ happen in the video? Please return its start and end times.

Table 7. Prompt templates for temporal grounding. We designed the templates of [1], [2], and [3] for Video-LLMs, as well as specific templates for GPT-40 and Gemini. For TimeChat, VTimeLLM, and VTG-LLM, we follow the official grounding prompts. We replace the q in the prompts with the query sentence.

ing tasks.

## **B.2. Evaluation Setup**

As previously mentioned in Section 4, the feasibility of general-purpose Video-LLMs and closed-source models for temporal grounding is absent. In this section, we provide adaptation details for these models.

**General-purpose Video-LLMs.** We empirically found that the general-purpose Video-LLMs are very sensitive to prompts, easily failing to extract timestamps from their answers. For instance, the model does not meet our expectations, such as answering "I'm unable to find timestamps in the video", or the model's answer only includes a single timestamp like "The given event happens at 3 seconds." We categorize these answers as *Invalid output*. Therefore, we carefully design several prompts for temporal grounding to find the stable setting for each model and provide experiment results in Table 8. For Video-LLaMA, we have two options for LLM backbones, Llama-2 and Vicuna. While using LLaMA-2 as an LLM backbone and the first prompt

Method	LLM	Prompt	Charad	es-CON	ActivityNet-CON			
		Trompt	R@1, 0.5	R@1, 0.7	R@1, 0.5	R@1, 0.7		
		[1]	15.8	5.7	9.6	2.0		
	Llama2-7B	[2]	0.2	0.1	0.1	0.1		
Video I LeMA		[3]	0.3	0.2	0.4	0.0		
VIUCO-LLaWIA		[1]	<u>14.2</u>	<u>4.1</u>	12.8	<u>3.1</u>		
	Vicuna-7B	[2]	8.4	1.5	16.8	5.3		
		[3]	3.4	0.3	2.8	1.0		
	Vicuna-13B	[2]	8.0	2.6	14.7	8.3		
		[1]		—— Invalia	output —			
Video-LLaVA	Vicuna-7B	[2]	9.6	3.9	<u>13.3</u>	<u>5.4</u>		
		[3]	<u>9.4</u>	<u>3.2</u>	13.4	6.5		
		[1]	12.3	6.8	3.0	0.3		
Video-ChatGPT	Vicuna-7B	[2]	<u>13.5</u>	<u>7.3</u>	3.4	<u>0.7</u>		
		[3]	14.4	7.6	<u>3.3</u>	0.8		
		[1]	7.2	2.1	10.5	4.9		
VideoChat2	Vicuna-7B	[2]		—— Invalia	output ———			
		[3]		Invalid output				
		[1]		Invalia	output —			
Video-LLaMA2	Mistral-7B	[2]	17.6	8.3	10.4	4.8		
		[3]	20.0	9.3	10.4	<u>4.7</u>		
VTG-LLM	Llama2-7B	-	26.0	11.0	6.8	3.0		
VTimeLLM	Vicuna-7B	-	27.3	11.4	31.9	17.0		
TimeChat	Llama2-7B	-	30.5	13.5	4.6	2.0		
GPT-40	-	-	28.5	12.3	26.8	12.0		
Gemini 1.5 Flash	-	-	34.6	16.4	37.8	26.9		

Table 8. Grounding performance of Video-LLMs across various experiment settings. The number in the Prompts column denotes the prompts in Table 7. Note that the results are obtained from our evaluation sets, not the original test split. The *Invalid output* represents when the model fails to include timestamps in its answer.

Method	# Frames	Charad	les-STA	ActivityNet-Captions		
		R@1, 0.5	R@1, 0.7	R@1, 0.5	R@1, 0.7	
	10	29.0	16.0	33.0	13.0	
GPT-40	32	28.0	10.0	23.0	<u>9.0</u>	
	64	13.0	9.0	18.0	5.0	

Table 9. Grounding performance of GPT-40 across different video frame inputs. For efficiency, we randomly sample 200 videos from Charades-STA and ActivityNet-Captions and find the acceptable grounding performance of GPT-40.

achieves the best grounding performance, it demonstrates unstable performance depending on the prompt. Therefore, we utilize Vicuna and the first prompt (*i.e.*, [1] in Table 7), demonstrating better consistency (see Table 13). Other Video-LLMs like Video-ChatGPT are fixed with a specific LLM backbone, so we follow their development settings. In conclusion, we use the first prompt for Video-LLAMA and VideoChat2, the second prompt for Video-LLAVA, and the third prompt for Video-ChatGPT and VideoLLaMA2.

**Time-aware Video-LLMs.** For TimeChat, VTimeLLM, and VTG-LLM, we follow the same prompt templates, which are used in their official instruction tuning for temporal grounding. Likewise, we use the official codes to extract the timestamps from their predictions.

**Closed-source Models.** Given the considerable budget required to utilize GPT-40, we first define the set of frame numbers {10, 32, 64} and sub-sample 100 videos from each dataset to efficiently find the optimal setting. In Table 9, we provide the grounding performance of GPT-40 across different frame numbers. Note that the performances in Table 9 are different from those in Table 1 because the number of test samples is different. During the experiment, we instruct GPT-40 on how many frames are in the given video and how long each frame is sampled. We find that using 10 frames shows the best performance. For Gemini, we use the official APIs to process video inputs.

**Visualization.** We present example question-answer pairs for all models in Table 15. The format of timestamps varies depending on the model, so we design a timestamp extraction function for each model.

#### **B.3.** Limitations of the Time-aware Models

We hypothesize that the excessive focus on temporal grounding with specialized elements in the time-aware models may limit its adaptability to various video understanding tasks and hinder generalization capabilities. For instance, TimeChat develops a time-aware frame encoder that explicitly encodes video features along with the corresponding sentences containing timestamps, such as "This frame is sampled at 2s." VTG-LLM utilizes digit time tokens, such as <TIME ZERO> and <TIME ONE>, to effectively represent timestamps in videos. While such methods may be effective for answering questions that require timestamps within videos, providing these timestamp priors could hinder generalization across various video understanding tasks. This limitation is particularly evident in VTG-LLM, which shows poor relative consistency for verification probes, and we empirically find that VTG-LLM often struggles to follow instructions. Specifically, VTG-LLM frequently generates timestamps as answers, even when asked for verification probes.

## **C. Details of Prompting Methods**

In this section, we provide details of the two prompting methods, Chain-of-Thought and Description prompting. **Chain-of-Thought.** We provide prompts that we utilize for each prompting method. In Figure 8, we give three chain

each prompting method. In Figure 8, we give three chainof-thought prompts that we design for each probe in our evaluation. The three tasks, temporal grounding, rephrased grounding, and shifted grounding, share the same prompt since their output is the same. For holistic and compositional verification, we instruct on which part the model should concentrate either a whole sentence or sub-elements.

**Description Prompting.** We first ask the model "Please describe the given video in detail." Then we keep the generated video description in chat history to perform the probes during our evaluation. We give a visual example of Video-ChatGPT using the description prompting method in Figure 9. In our previous discussion, we highlighted that while using the description prompting method enhances the model's consistency, it may lead to negative outcomes such as generating

Grounding / Rephrased Grounding / Shifted Grounding
Your task is to predict the start and end times of an action or event described by a query sentence based on the visual content of the video. Use Chain-of-Thought reasoning to break down the query, analyze key moments, and accurately identify the time range where the action occurs.
<ul> <li>### Chain-of-Thought Reasoning:</li> <li>1. **Step 1: Parse the Query**: Break down the query sentence to understand the key action or event that you need to locate.</li> <li>2. **Step 2: Analyze the Video Features**: Examine the sequence of video frames to detect patterns that match the key action described in the query.</li> <li>3. **Step 3: Identify the Temporal Boundaries**: Use temporal reasoning to find the start and end frames of the action based on the video features.</li> <li>4. **Step 4: Predict Start and End Times**: Map the identified frames to timestamps in the video, making sure the start and end times align with the query.</li> <li>5. **Step 5: Verify the Answer**: Check if the predicted time range accurately captures the action described in the query.</li> </ul>
Holistic Verification
You are a model designed to predict when specific events occur in a video based on a query sentence. Your task is to verify whether the event described in the query occurs in the given moment of the video.
<ul> <li>### Chain-of-Thought Reasoning:</li> <li>1. **Step 1: Verify the Event in the Predicted Time Range**: Analyze the video features from the predicted start time to the end time. Determine if the event described in the query occurs within this time range.</li> <li>- Example: For the query "The person is cooking," check for visual patterns such as a stove or kitchen utensils during the predicted moment.</li> <li>2. **Step 2: Answer the Verification Question**: Respond to the question:</li> <li>- **"Is the event '[event]' present from (start_time) to (end_time) seconds in the video?"**</li> <li>- Example: "Is the event 'The person is cooking' present from 30.0 to 40.0 seconds in the video?"</li> <li>- If find the event in the given moment, your answer should be "No.".</li> </ul>
Compositional Verification
You are a model designed to analyze the compositional elements of an event in a video. Your task is to verify whether each compositional element occurs during the given moment in the video based on the specific question you receive. Instead of analyzing the entire event at once, you will answer questions about individual components of the scene. ### Chain-of-Thought Reasoning:
1. **Step 1: Analyze the Video Features for the Specific Element**: Analyze the video features from the start time to the end time. Look for visual evidence of the specific compositional element described in the question. 2. **Step 2: Answer the Compositional Question**: Respond to the question:
- If you find a young girl in the given video moment, your answer should be "Yes.". If it is not present, your answer should be "No.".

Figure 8. Chain-of-Thought prompts for temporal grounding and the probes. Since temporal grounding, rephrased grounding, and shifted grounding share the same goal, predicting the video moment accurately, they use the same prompt. For verification probes, we specify which parts the model should focus on, whether it's the entire sentence or specific sub-elements.



Figure 9. The responses of Video-ChatGPT with Description prompting. While Video-ChatGPT generates a rich video caption, it contains incorrect information that may mislead accurate predictions. In this case, Video-ChatGPT fails to accurately predict the timestamp in the video.

incorrect descriptions or hallucinations. As illustrated in Figure 9, Video-ChatGPT provides a detailed description

of the video, but some of the information in the generated description is inaccurate. For instance, the video depicts two men in a gym, with only one of them wearing a red shirt. However, the model incorrectly states that both men are wearing red shirts. This might lead to inaccurate moment predictions.

## **D.** Details of Instruction Tuning

In this section, we provide a detailed experiment setup for instruction tuning and VTune.

**Instruction Tuning.** Following the previous work [31], we collect the annotations in the train split and convert the annotations into a task-specific template to derive Video-LLMs can predict the timestamps. For example, if the query "The person closes the laptop." is grounded in 0 to 5 seconds in the video, we prompt the models "Localize the visual content described by the given textual query 'The person closes the laptop.' in the video, and output the start and end timestamps in seconds.". Then the model's answer should be "The given query happens in 0 - 5 seconds." We utilize the official codes and configurations to conduct instruction tuning upon their official checkpoints.

**Event Temporal Verification Tuning.** We construct verification questions from the training split of each dataset.

Method		Charades-CON						ActivityNet-CON					
Nictiou	Ground	Ground (0.7)	R-Ground	R-Ground (0.7)	S-Ground	S-Ground (0.7)	Ground	Ground (0.7)	R-Ground	R-Ground (0.7)	S-Ground	S-Ground (0.7)	
Open-source (general-purpose)													
VideoChat2	7.2	2.1	5.4 (74.5)	4.4 (60.7)	1.0 (13.7)	0.3 (3.9)	10.5	4.9	8.7 (82.8)	7.9 (75.6)	0.6 (6.0)	0.1 (1.3)	
Video-LLaVA	9.4	3.2	7.6 (80.8)	7.4 (78.7)	2.8 (30.3)	0.4 (4.5)	13.4	6.5	10.0 (74.5)	<u>9.4</u> (70.3)	3.1 (23.0)	1.0 (7.3)	
Video-LLaMA	14.2	4.1	10.6 (74.9)	9.3 (65.6)	5.3 (37.6)	8.3 (5.9)	12.8	3.1	8.5 (66.8)	8.1 (63.9)	7.2 (56.8)	1.6 (12.5)	
Video-ChatGPT	14.4	7.6	12.8 (89.2)	12.7 (87.9)	1.3 (8.8)	0.4 (2.9)	3.3	0.8	2.8 (84.0)	2.7 (82.6)	0.1 (4.1)	0.1 (4.1)	
Video-LLaMA2	20.0	9.3	16.8 (83.8)	16.6 (83.1)	3.8 (19.0)	1.1 (5.6)	10.4	4.7	8.2 (78.6)	7.4 (70.9)	1.5 (14.8)	0.3 (2.7)	
Open-source (time-aware)													
VTG-LLM	26.0	11.0	16.1 (62.1)	11.4 (43.8)	8.3 (32.0)	2.5 (9.7)	6.8	3.0	5.3 (78.0)	3.1 (45.0)	0.2 (3.0)	0.1 (1.0)	
VTimeLLM	27.3	11.4	22.7 (83.2)	22.2 (81.3)	7.3 (26.9)	2.2 (8.2)	31.9	17.0	26.4 (82.7)	25.0 (78.5)	8.9 (27.9)	3.6 (11.2)	
TimeChat	30.5	13.5	25.0 (82.1)	20.2 (66.2)	5.6 (18.5)	1.3 (4.1)	4.6	2.0	2.9 (64.1)	2.7 (58.5)	1.0 (21.2)	0.3 (7.5)	
Closed-source													
GPT-40	28.5	12.3	21.2 (74.3)	12.7 (44.6)	9.3 (32.8)	2.8 (9.9)	26.8	12.0	18.1 (67.5)	12.9 (48.3)	10.4 (38.8)	3.4 (12.5)	
Gemini 1.5 Flash	34.6	16.4	29.7 (85.7)	24.8 (71.7)	24.8 (71.7)	13.5 (39.1)	37.8	26.9	30.8 (81.4)	27.6 (72.9)	24.8 (65.6)	14.8 (39.2)	

Table 10. Grounding and consistency results of models with different IoU thresholds. The number in brackets represents the IoU threshold.

Hyper-parameter	Instru	ction Tuning	Event Temporal Verification Tuning		
,	Charades-STA	ActivitNet-Captions	Charades-STA	ActivitNet-Captions	
Epochs	3	3	3	1/3	
Batch size	1	1	1	1	
Learning rate	3e-5	3e-5	3e-5	3e-5	
Warm-up learning rate	1e-6	1e-6	1e-6	1e-6	
Iters-per-epochs	3102	24811	9332	51377	
Warmup-steps	3102	14916	9332	25688	

Table 11. Hyper-parameters for each tuning method. While we train 3 epochs for Video-LLaMA with VTune on ActivityNet-Captions, we empirically find that TimeChat only needs 1 epoch for training. Except for this case, we use the same hyper-parameters in both models.

Specifically, we generate both aligned and misaligned sentences from the original query and also generate correction information on what information is wrong in misaligned sentences by providing the prompt in Figure 11 to GPT-4o-mini. Then we pair aligned sentences with affirmative responses and misaligned sentences with denial responses and their corresponding corrections.

**Experiment Settings.** The number of queries by type for each dataset can be found in Table 12. For VTune, we follow the official configurations and only change the parameters, including iters per epoch and warmup steps, to reflect the number of annotations for each training. We report hyperparameters used in each instruction tuning method in Table 11. We use  $4 \times 80$  GB A100 GPUs for all experiments and use the same prompts and hyper-parameters in both Video-LLaMA and TimeChat. In our setting, Video-LLaMA requires less than one day, and TimeChat requires at least two days for instruction tuning. For VTune, Video-LLaMA takes under two days, while TimeChat requires six days.

## **E. Additional Experiments**

In this section, we provide additional experiment results to further provide insights. We first provide the consistency evaluation results using different IoU values. Then we examine how the prompt for temporal grounding and scalability of LLM impact consistency. Then we further analyze the results of rephrased grounding and shifted grounding.

Dataset	G	Е	Т	Total
Charades-STA	12408	76145	10691	99244
ActivityNet-Captions	37330	138340	29840	205510

Table 12. The number of grounding and verification queries in each dataset. The following columns represent a specific query type used: G for Grounding, E for Event Verification, and T for Temporal Verification.

**Consistency Results across Different IoU thresholds.** In Table 10, we provide additional experiment results using an additional IoU threshold (*i.e.*, 0.7) for temporal grounding and grounding probes.

The impact of Temporal Grounding Prompt on Consistency. To investigate this, we provide the results in Table 13. As the prompt designs affect the grounding performance of Video-LLMs, we also investigate their consistency. Specifically, we select a series of settings that demonstrate high performance for each model in Table 8. While Video-LLaMA with Vicuna-7B and the second prompt demonstrate better relative consistency scores for verification probes, there is no significant performance gap compared to other settings. Likewise, for Video-LLaVA, Video-ChatGPT, and Video-LLaMA2, the grounding prompt does not significantly impact the model's performance.

The impact of Scalability of LLM on Consistency. Additionally, we evaluate Video-LLaMA with Vicuna-13B and confirm that there are no significant improvements. Specifically, while it shows higher relative consistency in rephrased grounding, it fails to demonstrate strong capabilities for other probes. Likely, a larger-scale LLM does not necessarily represent improved consistency. Unfortunately, we cannot evaluate the models with larger-scale LLM backbones, as most of them have either conducted experiments only with 7B LLMs or have not released the corresponding checkpoints. For instance, TimeChat and VTG-LLM haven't officially conducted experiments using 13B LLM backbones. VTimeLLM reports the performance using Vicuna-13B in the original paper, but has not officially released the corre-

Method	LLM	Prompt	Prompt	# Frames			Charades-C	ON			A	ctivityNet-C	ON	
Methou	EEM	Trompt	" I Tunics	Ground	R-Ground	S-Ground	H-Verify	C-Verify	Ground	R-Ground	S-Ground	H-Verify	C-Verify	
	Llama2-7B	[1]	8	15.8	12.8 (81.2)	<u>5.2</u> (33.2)	<u>7.2</u> (46.2)	<b>7.9</b> (50.3)	9.6	8.2 (85.7)	4.6 (48.7)	4.8 (50.8)	4.8 (50.7)	
Video-LLaMA	Vicuna-7B	[1]	8	14.2	10.6 (74.9)	<b>5.3</b> (37.6)	7.5 (53.3)	7.3 (51.7)	12.8	8.5 (66.8)	7.2 (56.8)	<u>7.3</u> (57.5)	<u>7.5</u> (58.9)	
	Vicuna-7B	[2]	8	8.4	5.9 (70.0)	2.9 (35.0)	4.5 (54.0)	4.6 (55.0)	16.8	14.5 (86.6)	9.1 (54.1)	9.8 (58.2)	9.9 (59.1)	
	Vicuna-13B	[2]	8	8.0	6.8 (85.9)	0.2 (3.5)	3.8 (48.3)	3.9 (49.5)	<u>14.7</u>	<u>12.8</u> (87.4)	1.0 (7.1)	6.5 (44.6)	7.1 (48.4)	
Video I LeVA	Vicuna-7B	[2]	8	9.6	<b>8.0</b> (84.3)	2.4 (25.0)	4.8 (50.8)	4.8 (50.3)	13.3	10.4 (78.6)	<b>3.8</b> (28.9)	<b>6.4</b> (48.4)	6.8 (51.5)	
VIUCO-LLaVA	Vicuna-7B	[3]	8	9.4	7.6 (80.8)	<b>2.8</b> (30.3)	<b>5.0</b> (52.8)	4.7 (50.0)	13.4	10.0 (74.5)	3.1 (23.0)	6.3 (46.9)	<b>7.0</b> (52.3)	
Vidao ChatCDT	Vicuna-7B	[2]	100	13.8	12.0 (87.5)	1.1 (8.3)	<b>6.6</b> (48.5)	6.9 (50.2)	3.4	2.7 (80.9)	<b>0.1</b> (4.0)	<b>1.7</b> (51.9)	1.7 (51.6)	
video-Cliator I	Vicuna-7B	[3]	100	14.4	12.8 (89.2)	1.3 (8.8)	6.5 (44.8)	<b>7.2</b> (50.0)	3.3	<b>2.8</b> (84.0)	<b>0.1</b> (4.1)	1.7 (50.4)	1.6 (49.2)	
Video LLoMA2	Mistral-7B	[2]	8	17.6	14.7 (83.7)	3.5 (20)	9.1 (52.1)	9.2 (52.8)	10.4	7.8 (75.6)	1.4 (13.5)	5.3 (51.5)	5.6 (54.1)	
viuco-LLawiA2	Mistral-7B	[3]	8	20.0	16.8 (83.8)	<b>3.8</b> (19.0)	<b>10.3</b> (51.5)	<b>10.6</b> (52.9)	10.4	<b>8.2</b> (78.6)	<b>1.5</b> (14.8)	<b>5.4</b> (52.4)	<b>5.7</b> (54.7)	

Table 13. Consistency evaluation of Video-LLMs across different settings. While the model's performance varies across different prompts for temporal grounding and LLM backbones, there are no significant differences between them.

Method	(	Charad	es-COl	N	ActivityNet-CON				
	WR	AP	WO	Avg	WR	AP	WO	Avg	
VideoChat2	76.2	75.3	75.7	75.7	86.0	84.7	81.0	83.9	
Video-LLaVA	83.7	74.7	<u>75.5</u>	77.9	76.4	70.0	69.7	72.0	
Video-LLaMA	74.3	75.0	65.1	71.4	89.1	<u>85.3</u>	85.2	86.5	
Video-ChatGPT	92.8	86.6	90.8	90.0	88.5	88.5	88.8	88.6	
Video-LLaMA2	88.0	82.0	82.8	84.2	82.6	<u>71.3</u>	69.9	74.6	
VTG-LLM	56.8	<u>53.8</u>	53.8	54.8	77.2	60.4	61.3	66.3	
VTimeLLM	77.3	74.9	73.4	75.2	63.3	62.8	63.3	63.1	
TimeChat	83.8	84.7	80.4	82.9	84.4	<u>80.5</u>	78.6	81.1	
GPT-40	60.0	64.7	<u>61.9</u>	62.2	56.9	58.5	<u>57.3</u>	57.5	
Gemini 1.5 Flash	75.4	77.4	74.9	75.9	76.0	<u>71.4</u>	71.4	72.9	

Table 14. Rephrased grounding results across three modification rules. WR: Word Replacement, AP: Active-to-Passive conversion, and OC: Word Order change. We report mIoU for rephrased grounding. Most models generally achieve a high score for WR.

#### sponding checkpoint.

Analysis of Rephrased Grounding. In Table 14, we provide the performance for rephrased grounding by three modification rules. The rules are previously introduced in Section 3.1 including word replacement, active-to-passive conversion, and word order change. We confirm that most models temporally ground rephrased queries with word replacement. As we discussed in Section 4, this suggests the need for improvements in consistent grounding and demonstrates the effectiveness of our modification rules providing various language variations.

**How does VTune improve grounding?** We speculate that VTune enhances grounding in multiple ways. Fig 10-(A) shows that verification tuning alone improves grounding performance by approximately 20% (from an initial 30.5), suggesting that verification tasks inherently benefit grounding. This is likely because verification tasks are more general and align with the prior QA capabilities of LLMs, whereas fine-tuning for grounding may override this knowledge due to requiring task-specific timestamp templates, resulting in poor consistency. Additionally, Fig 10-(B) highlights the importance of dataset scale and diversity, indicating that verification queries serve as effective data augmentation, ex-



Figure 10. Experiments on Charades-CON with TimeChat. (A): Performance evaluation across different ratios of event and temporal verification queries. The total queries are fixed at 12k, matching the # of grounding queries in the original dataset. (B): Performance evaluation across varying verification query scales. 100% scale corresponds to the performance of VTune reported in the paper. The number of unique words is a proxy for dataset diversity.

panding linguistic coverage. However, excessive training may diminish performance.

## F. Prompts for GPT

In this section, we provide detailed prompts that are utilized in our study.

- 1. We use the prompt in Figure 12 to generate aligned, misaligned, and compositional queries.
- 2. We use the prompt in Figure 13 to evaluate the model's answer for our probes.

You are an intelligent chatbot tasked with generating and analyzing sentences based on video content. Each input sentence describes a specific moment in the video. Your goals are as follows:

Generate Aligned and Misaligned Sentences:

Both aligned and misaligned sentences should be natural and realistic, without introducing any implausible or unrealistic information. If you find that the generated sentences using the above techniques are unnatural, you may skip generating them. Ensure that generate at least three aligned and misaligned sentences and the number of aligned and misaligned sentences should be equal.

Aligned Sentences (A): 1. Replace Words: Replace key nouns and verbs while maintaining the sentence's meaning.

2. Active to Passive: Convert active sentences to passive voice.

3. Word Order Changes: Rearrange the word order while preserving the original meaning.

Misaligned Sentences (M): Alter the meaning so that the sentence is no longer releavnt to the original context.

Correction for Misaligned Sentences (C): For each misaligned sentence, generate a question that highlights the difference from the input sentence. Provide an answer that explains why the misaligned sentence is incorrect and offers the correct version. Avoid using phrases like "the sentence does not mention," "the sentence does not specify," or "the sentence suggests." Instead, think the input

sentence as a video content.

The output should be formatted as a Python dictionary with the following structure: Input sentence: "A woman is reading a book while sitting under a tree. Output: { "A": [ "A woman is perusing a text while resting under a tree.", "A book is being read by a woman while she is sitting under a tree.", "While sitting under a tree, a woman is reading a book." ], "M": [ "A man is throwing a ball while standing near a building.", "A woman is playing with a toy while lying on the grass"

"While standing under a tree, a woman is reading a book.",

], "C": [

"We cannot see a man throwing a ball. We can see a woman reading a book.",

"We cannot see both a man and toy in the given video moment. We can see a woman reading a book while sitting under a tree.",

- "A woman is sitting under a tree, not standing." 1
- }

Figure 11. The prompt to generate verification questions for VTune.

You are an intelligent chatbot designed for generating and decomposing sentences. You are an intelligent chatbot designed for generating and decomposing sentences. Your task is to generate aligned and misaligned sentences, as well as compositional information, based on the input sentence. Ensure that the generated sentences stay within the context of the provided sentence without introducing any new information.

INSTRUCTIONS: Both aligned and misaligned sentences should be natural and realistic, without introducing any implausible or unrealistic information. If you find that the generated sentences using the above techniques are unnatural, you may skip generating them. Ensure that generate at least three aligned and misaligned sentences and the number of aligned and misaligned sentences should be equal.

- Aligned Sentences (A): Follow below tecniques.

- 1. Replace Words: Replace key nouns and verbs while maintaining the sentence's meaning.
- 2. Active to Passive: Convert active sentences to passive voice.
- 3. Word Order Changes: Rearrange the word order while preserving the original meaning.

- Misaligned Sentences (M): Detect the main components in the given query and then alter the meaning so that the sentence is no longer relevant to the original context.

```
- Compositional Information (C): Your task is analyze the given query sentence and decompose it into its fundamental components:
```

- 1. Subject: Identify the primary entities or characters mentioned in the sentence.
- 2. Action: Describe the actions being performed by the subjects or what is happening to them.
- 3. Relation: Determine the relationship between the subjects and other elements in the sentence, such as objects, locations, or other entities.

Based on the extracted components, create a series of yes/no questions:

- 1. Positive Questions (Y): Ensure these questions are directly aligned with the original sentence, capturing its true meaning.
- 2. Negative Questions (N): Modify the sentence meaningfully to generate misaligned questions, ensuring the correct answer is 'no.'. Ensure that all generated questions are coherent, natural, and contextually appropriate.

The output should be formatted as Python dictionary style as follows in example:

Input sentence: "She is surrounded by two other accordions as she instructs on how to play the instrument." Output: { "A": [ "She is encircled by two other accordions while she teaches how to play the instrument.", "As she instructs on how to play the instrument, two other accordions surround her.", "Two accordions are near her as she gives instructions on playing the instrument.", "M": [ "M": [ "M": [ "An old man is alone on stage and instructs how to play the issurrounded by two kids and instructs them.", "C": { "Y": ["Is she surrounded by accordions?", "Does she instruct on playing the instrument?"], "Y": ["Is she surrounded by guitars?", "Does she assemble instruments?"], } Now it's your turn. Please write the answer based on the given sentence.

Flease while the answer based on the given semence.

DO NOT PROVIDE ANY OTHER OUTPUT TEXT OR EXPLANATION. Only provide the Python dictionary string. Input sentence: "{sentence}"

Output:

Figure 12. The prompt for generating aligned, misaligned, and compositional queries.

You are an intelligent evaluator tasked with determining the correctness of an answer in response to a given question. Based on the expected response type (positive or negative), you need to decide whether the provided answer aligns correctly with both the content and timestamps in the question.

INSTRUCTIONS:

1. Input Information:

Question: A question about a specific moment or event in the video.

- Expected Answer Type:

"pos" (Positive): The expected answer should agree with the content of the question. It may include phrases like "Yes," "Correct," or similar content that confirms the question's scenario.

- "neg" (Negative): The expected answer should disagree with the content of the question. It may include phrases like "No," "Incorrect," or similar content that refutes the question's scenario.

- Provided Answer: The actual answer given to the question. You must evaluate whether it is correct based on the question and the expected answer type.

2. Evaluation Process:

- If the expected answer type is "pos," the provided answer should be a positive affirmation of the question.
- If the expected answer type is "neg," the provided answer should refute or disagree with the question.
- Contextual Consistency: If the provided answer contains a phrase like "Yes," but mentions a different time range or contradicts the actual event happening in the video, it should be marked incorrect.

- Any variation in timestamps in the provided answer (such as mentioning events outside the specified timestamp window) is considered incorrect. - If the provided answer matches both the expected answer type and the timestamps, return 1 (correct).

- If the provided answer contradicts or is not relevant to the expected answer type, timestamps, or context, return 0 (incorrect).

3. Output format:

The output should be formatted as a dictionary format with the key "result", and the value being either:

- 1 if the answer is correct according to the question, expected answer type, and timestamps

- 0 if the answer is incorrect.

EXAMPLES:

Input:

- Question: Does the event 'A young woman is standing and speaking with her friends' from 15 to 25 seconds in the video?

- Expected Answer Type: neg

- Provided Answer: No, we can not see a young woman standing from 15 to 25 seconds in the video.

Output:

"result": 1

}

Input:

- Question: Is the man reading a book under the tree from 0 to 30 seconds in the video?
- Expected Answer Type: pos
- Provided Answer: Yes, the man is reading a book from 0 to 30 seconds in the video.

Output:

"result": 1 }

Input:

- Question: Does the event 'A man play soccer' from 30 to 60 seconds in the video?

- Expected Answer Type: pos

- Provided Answer: The man is playing soccer from 80 to 90 seconds in the video.

Output:

"result": 0 }

Input:

- Question: Does the event 'Person opens the window a few more times.' not happen from 0.0 to 5.0 seconds in the video? Please answer with 'Yes' or 'No'.

- Expected Answer Type: pos

- Provided Answer: Yes, the event 'Person opens the window a few more times' happens from 10.0 to 15.0 seconds.

Output:

}

{ "result": 0 Now it's your turn. Input: - Question: {question} Expected Answer: {expected}

- Provided Answer: {answer}

PLEASE DO NOT PROVIDE ANY OTHER OUTPUT TEXT OR EXPLANATION. ONLY PROVIDE A DICTIONARY FORMAT.

Output:

Figure 13. The prompt to assess the correctness of the model's answer for verification probes. Specifically, we ask GPT-4o-mini to assess whether the answer aligns with the expected answer.

Method	Question-Answer pairs
Video-LLaMA	<ul><li>Q: When does the event 'person puts bag of groceries on table.'? Please only return its start time and end time.</li><li>A: The event 'person puts bag of groceries on table' happens at 0.0 seconds. It lasts for 19.2 seconds.</li><li>Q: Is the person running into the room from 0.0 to 19.2 seconds in the video?</li><li>A: Yes.</li></ul>
Video-LLaVA	<ul> <li>Q: Please answer when the event 'He keeps adding paint along the way, sometimes directly to the canvas.' occurs in the video. The output format should be: 'start - end seconds'. Please return its start time and end time.</li> <li>A: 0:45 - 0:65</li> <li>Q: Does the event 'man is taking off a violin from the case and gives it to a man.' not happen from 0.0 to 36.0 seconds in the video?</li> <li>A: Yes, the event 'man is taking off a violin from the case and gives it to a man' does not happen from 0.0 to 36.0 seconds in the video.</li> </ul>
Video-ChatGPT	<ul> <li>Q: Please answer when the event 'The man then instructs the man holding the rope to pull the row down a few times and he's talking the whole time.' occurs in the video. The output format should be: 'start - end seconds'. Please return its start time and end time.</li> <li>A: The event 'The man then instructs the man holding the rope to pull the row down a few times and he's talking the whole time.' occurs at 0:15 - 0:30.</li> <li>Q: Is the event 'The man subsequently instructs the individual holding the rope to draw the row down multiple times while he's speaking continuously.' present from 15 to 30 seconds in the video.</li> </ul>
Video-Chat2	<ul><li>Q: When does the event 'person closes the door a few more times.' occurs in the video? Please return its start time and end time in seconds. The output format should be: 'start - end seconds'.</li><li>A: The event 'person closes the door a few more times' occurs in the video at the start time of 00:00 and ends at 00:02.</li><li>Q: Is the event 'The individual shuts the door several additional times.' occurring from 0 to 2 seconds in the video?</li><li>A: The event of the individual shutting the door several times is happening from 0 to 2 seconds in the video.</li></ul>
Video-LLaMA2	<ul> <li>Q: Please answer when the event 'A person is seen knitting close up and leads into a woman speaking.' occurs in the video. The output format should be: 'start - end seconds'. Please return its start time and end time.</li> <li>A: The event 'A person is seen knitting close up and leads into a woman speaking.' takes place from 0 seconds to 11.9 seconds.</li> <li>Q: Is the event 'Close up, a person is observed knitting, leading into a woman who is speaking.' not present from 0 to 11.9 seconds in the video?</li> <li>A: Yes, the event 'Close up, a person is observed knitting, leading into a woman who is speaking.' is not present from 0 to 11.9 seconds in the video.</li> </ul>
TimeChat	<ul> <li>Q: Please find the visual event described by a sentence in the video, determining its starting and ending times. The format should be: 'The event happens in the start time - end time'. For example, The event 'person turn a light on' happens in the 24.3 - 30.4 seconds. Now I will give you the textual sentence: 'person starts to run out of the room.' Please return its start time and end time.</li> <li>A: The given query happens in 20 - 25 seconds.</li> <li>Q: Is the event 'Out of the room, the person starts to run.' occurring from 20.0 to 25.0 seconds in the video? Please answer with 'Yes' or 'No'.</li> <li>A: Yes.</li> </ul>
VTimeLLM	<ul> <li>Q: During which frames can we see the event 'The man then instructs the man holding the rope to pull the row down a few times and he's talking the whole time.'?</li> <li>A: The event 'The man then instructs the man holding the rope to pull the row down a few times and he's talking the whole time.' can be seen from 17 to 71.</li> <li>Q: Is the event 'The woman then instructs the man holding the rope to pull the row up a few times and she's silent the whole time.' occurring from 17 to 71 frames in the video?</li> <li>A: Yes, that's correct.</li> </ul>
VTG-LLM	<ul> <li>Q: Find the video segment that corresponds to the given textual query 'person puts a bag on the sofa.' and determine its start and end seconds.</li> <li>A: 0020.0 - 0025.0 seconds, person puts a bag on the sofa.</li> <li>Q: Is the event 'person puts a bag on the sofa' occurring from 20.0 to 25 seconds in the video? Please answer with 'Yes' or 'No'.</li> <li>A: 0020.0 - 0025.0 seconds, person puts a bag on the sofa.</li> </ul>
GPT-40	<ul> <li>Q: The video consists of 10 frames, with a total duration of 153.35 seconds. Each frame is sampled every 15.0 seconds. In which time frames does the event He mixes some pain and paints part of a canvas, then does another color. happen? Identify the frames where the event takes place and provide the answer like start - end seconds For example, if you think the given event occurs from 5 to 10 seconds in the video, please return the 'The given happens in 5 - 10 seconds.' Please only include the timestamp in the answer without any other content.</li> <li>A: The given event happens in 0 - 60 seconds.</li> </ul>
Gemini 1.5 Flash	<ul><li>Q: When does the event 'a person walks through a doorway.' happen in the video? Please return its start and end times.</li><li>A: The person starts walking through the doorway at 00:07 and finishes at 00:24.</li><li>Q: Is the event 'a person walks through a doorway.' included from 7 to 24 seconds in the video? Please answer with Yes or No.</li><li>A: Yes.</li></ul>

Table 15. Example question-answer pairs of the models during our evaluation.