CoTasks: Chain-of-Thought based Video Instruction Tuning Tasks

Anonymous ACL submission

Abstract

Despite recent progress in video large language models (**VideoLLMs**), a key open challenge remains: how to equip models with chain-of-thought (**CoT**) reasoning abilities grounded in fine-grained object-level video understanding. Existing instruction-tuned models, such as the Qwen and LLaVA series, are trained on high-level video-text pairs, often lacking structured annotations necessary for compositional, step-by-step reasoning. We propose **CoTasks: Chain-of-Thought based Video Instruction Tuning Tasks**, a new framework that decomposes complex video questions of existing datasets (*e.g.*, NeXT-QA, STAR) into four entity-level foundational tasks: frame localization, entity tracking, spatial and temporal relation extraction. By embedding these intermediate CoT-style reasoning steps into the input, CoTasks enables models to explicitly perform object-centric spatiotemporal reasoning. Experiments on the NeXT-QA benchmark show that CoTasks significantly enhance inference performance: LLaVA-video-7B improves by **+3.3** points in average GPT-4 evaluation score, and Qwen2.5-VL-3B gains **+17.4**, with large boosts in causal (**+14.6**), temporal (**+10.9**) and descriptive (**+48.1**) subcategories. These results demonstrate the effectiveness of CoTasks as a structured CoT-style supervision framework for improving compositional video reasoning.

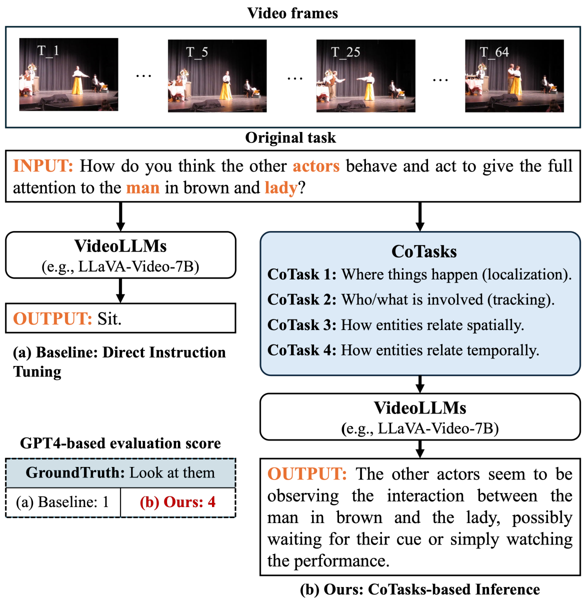

Figure 1: Comparison of LLaVA-video-7B with and without CoTasks. Given a complex video question, direct inference yields a shallow answer ("Sit"), whereas CoTasks guides the model through entity-aware reasoning, resulting in a more grounded and descriptive response.

1 Introduction

Video Large Language Models (VideoLLMs) are rapidly gaining importance in applications such as *interactive video QA systems*, including ChatGPT (OpenAI, 2023) and Gemini (DeepMind, 2023), and *embodied agents*, including OK-Robot (Liu et al., 2024c) and DynaMem (Liu et al., 2024b). These models aim to understand complex, dynamic visual scenes and generate coherent answers or decisions from high-level natural language queries. Despite recent advances, a key open challenge remains: how to equip these models with structured reasoning abilities that reflect the step-by-step understanding humans apply when interpreting real-world videos.

Current state-of-the-art VideoLLMs, including the Qwen (Bai et al., 2023; Wang et al., 2024; Bai et al., 2025) and LLaVA (Liu et al., 2023; Zhang et al., 2024) families, are predominantly trained using high-level video instruction tuning. These models perform reasonably well on surface-level tasks, but often fail in scenarios that require nuanced object-level and temporal reasoning. As shown in Figure 1, given a complex video-based question, models like LLaVA-video-7B tend to produce shallow answers such as "Sit" failing to ground the response in the visual context or reflect the full narrative. We hypothesize that this limitation arises from the absence of *intermediate supervision signals*, which guide the model to reason about **where**, **who/what**, and **how** entities interact over time. Inspired by Chain-of-Thought (CoT) prompting (Wei et al., 2022), we argue that injecting structured, step-wise visual cues can bridge this gap.

To this end, we introduce **CoTasks**, a new framework for *Chain-of-Thought based Video Instruction Tuning Tasks*. CoTasks decomposes high-level questions from existing video QA datasets (*e.g.*, NeXT-QA (Xiao et al., 2021), STAR (Wu et al., 2021)) into four foundational reasoning subtasks: (1) *frame localization*, (2) *entity tracking*, (3) *spatial relation extraction*, and (4) *temporal relation extraction*. By embedding these intermediate CoT-style reasoning steps into the input, CoTasks enables models to explicitly perform object-centric spatiotemporal reasoning. As a result, giving a high-level question, the model can ground and compose visual evidence step-by-step before producing an answer. As visualized in Figure 1, our approach transforms the vague output of standard VideoLLMs into a much richer response that closely mirrors human understanding.

We evaluate CoTasks using three VideoLLMs, *Qwen2.5-VL-3B*, *Qwen2.5-VL-7B*, and *LLaVA-video-7B* on the NeXT-QA and STAR benchmarks. To assess the upper bound of performance gains, we prompt models with the original question augmented by the ground-truth answers of the CoTasks subtasks. This allows us to isolate how much structured CoT-style context can contribute to final answer quality. Our inference-time augmentation strategy requires no architectural changes or re-training. CoTasks significantly boosts performance across all models. Notably, Qwen2.5-VL-3B—a lightweight model—achieves a **+17.4** point gain in average GPT-4 evaluation score, with large improvements in *causal* (**+14.6**), *temporal* (**+10.9**), and *descriptive* (**+48.1**) reasoning categories. These results demonstrate the effectiveness of CoTasks as a lightweight yet powerful mechanism for enhancing structured video reasoning. Moreover, our findings highlight the promise of using CoTasks as an instruction-tuning curriculum for building high-performance, resource-efficient VideoLLMs—a key requirement for on-device, real-world applications.

2 Problem setup

In this work, we focus on constructing a video instruction tuning dataset grounded in chain-of-thought (CoT) reasoning. Given object-centric video question answering (VideoQA) tasks such as NeXT-QA and STAR—which provide fine-grained annotations including object bounding boxes, intra-frame spatial relations, and inter-frame temporal relations—we reformulate the original multiple-choice questions into open-ended, free-form answering tasks. Furthermore, we augment each complex visual reasoning question with four foundational CoTasks: frame localization, object tracking, spatial relation extraction, and temporal relation extraction. The primary objective is to investigate whether these CoTasks can enhance the reasoning capabilities of state-of-the-art videoLLMs.

3 Related work

Multimodal Large Language Models (MLLMs). MLLMs, such as LLaVA (Liu et al., 2023), Qwen2-VL (Bai et al., 2023), and LLaMA3-Vision (meta, 2024), have extended language models into visual domains by integrating (1) a vision encoder (Radford et al., 2021; Tschannen et al., 2023) to extract features from images or video frames, (2) a projection module to align visual features with the language embedding space, and (3) a language model backbone for multimodal reasoning. Recent works, including Video-ChatGPT (Maaz et al., 2024), LLaVA-Video (Zhang et al., 2024), Qwen2.5-VL (Bai et al., 2025), and AdaReTaKe (Wang et al., 2025), extend MLLMs to videos by compressing and aligning frame sequences to enable long-context understanding. However, these models primarily focus on high-level instruction tuning without decomposing complex queries into structured sub-tasks—such as frame localization, object tracking, and spatiotemporal relation reasoning—limiting their ability to handle detailed object interactions or causal inference in dynamic scenes. In contrast, our proposed **CoTasks** framework addresses this limitation by decomposing high-level video QA into four foundational sub-tasks: entity-based frame localization, object tracking, spatial relation extraction, and temporal relation extraction. These sub-tasks serve as intermediate, Chain-of-Thought (CoT)-style reasoning steps that inject structured guidance into video instruction tuning and enhance fine-grained spatiotemporal reasoning.

Video Instruction Tuning dataset. Several large-scale video instruction datasets have recently been proposed to improve VideoLLMs' ability to follow complex visual-language instructions. LLaVA-Video-178K (Zhang et al., 2024) is a synthetic dataset of 178K video samples, each annotated with detailed captions and open- and closed-ended questions. Generated using GPT-4 and human feedback, it supports instruction tuning for models like LLaVA-Video but focuses on high-level annotations without explicit intermediate reasoning structures. VideoInstruct-100K (Maaz et al., 2024) consists of 100K video-instruction pairs with spatial, temporal, and reasoning-oriented questions, used to train Video-ChatGPT. While high-quality, it lacks structured object-centric decomposition, making it difficult to model fine-grained spatiotemporal reasoning. Video-STaR (VSTaR-1M) (Zohar et al., 2025) leverages diverse video sources (e.g., Kinetics-700 (Carreira et al., 2019), STAR (Wu et al., 2021)) to scale to 1M instruction pairs and aims to support general reasoning including CoT. However, it relies on task-level supervision without sub-task grounding. In contrast, our proposed CoTasks framework explicitly decomposes complex video QA into four structured sub-tasks—frame localization, object tracking, spatial and temporal relation extraction—which are formatted as intermediate Chain-of-Thought (CoT) steps. This object-level supervision enhances compositional video reasoning and better aligns with human-like step-by-step understanding, addressing a key limitation in prior datasets.

Chain-of-Thought Video Reasoning. Recent efforts have aimed to enhance VideoLLMs with step-by-step reasoning via Chain-of-Thought (CoT) prompting (Wei et al., 2022). LLaVA-o1 (Liu et al., 2024a) introduces CoT-based vision-language instruction tuning, demonstrating improvements in visual question answering with structured step generation. Similarly, works such as Agent-of-Thoughts (Chen et al., 2024a), Video-of-Thought (Zhang et al., 2023), and Visual CoT (Shao et al., 2024) explore CoT through modular reasoning agents, dynamic scene graphs, or benchmark-driven evaluation. SlowFocus (Nie et al., 2024) and MVU (Chen et al., 2024b) incorporate fine-grained temporal and object-level cues to improve interpretability. However, most approaches depend on implicit supervision or architectural adaptations and lack systematic object-level decomposition aligned with CoT stages. In contrast, **CoTasks** proposes four interpretable and structured sub-tasks—frame localization, entity tracking, and spatial/temporal relation extraction—that act as intermediate reasoning steps for improving CoT-based video instruction tuning.

4 Approach

4.1 CoTasks Definition

Consider the original task defined as follows:

Original Task: Comprehensive Video Understanding

This task requires answering a high-level video question that demands object-level context and comprehensive scene understanding.

Question Format: Open-ended question requiring holistic understanding of the video content.
Answer Format: Free-form short text.

We introduce a set of auxiliary tasks termed **CoTasks**. Each CoTask targets a distinct aspect of object-centric visual reasoning grounded in video understanding. We define the four foundational **CoTasks** as follows:

CoTask 1: Object-based Frame Localization

This task involves identifying video frames where a specific combination of objects co-occur, as specified in the original question.

Question Format: Ground entities and identify frames matching context in the target question.
Answer Format: A dictionary with the following keys: `{"objects": [str], "timestamps": [int]}`

CoTask 2: Object Tracking with Bounding Boxes

This task requires tracking and localizing each mentioned object by generating bounding boxes over relevant video frames.

Question Format: Get object locations (bounding boxes) in frames listed in CoTask 1.
Answer Format: A list of dictionaries, each with: `[{"frame": int, "objects": [{"label": str, "bbox": [x1, y1, x2, y2]}, ...]}, ...]`

4.2 CoTasks construction pipeline

Figure 2 illustrates the process of constructing CoTasks by decomposing a given video question answering task (*e.g.*, from NeXT-QA) into four structured CoTasks using object-level annotations (*e.g.*, VidOR (Shang et al., 2019)).

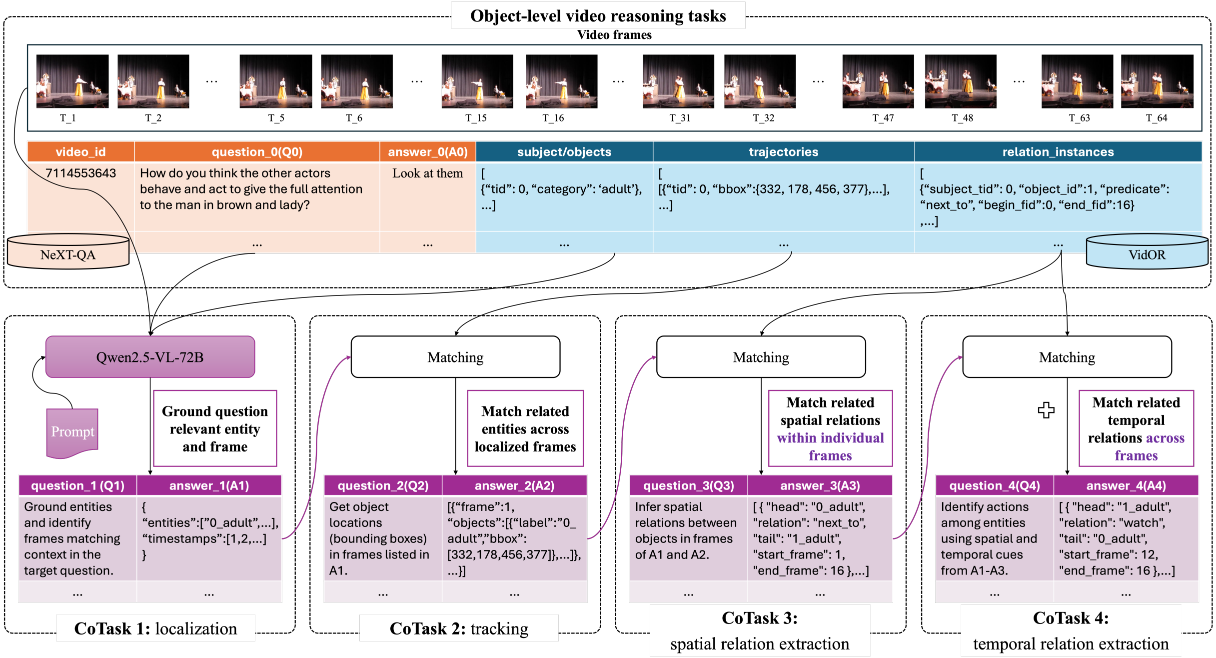

Figure 2: The figure illustrates the process of decomposing a VideoQA task (*e.g.*, from NeXT-QA) into four structured CoTasks using object-level annotations (*e.g.*, from VidOR). First, we reconstruct the VideoQA task to align video frames with object-level annotations (§ 4.2.1). Then, we construct CoTasks based on the reconstructed VideoQA, where each CoTask builds upon the previous one (§ 4.2.2).

CoTask 3: Spatial Relation Extraction

This task involves extracting spatial relationships (*e.g.*, `next_to`, `in_front_of`) between pairs of objects within a frame span.

Question Format: Infer spatial relations between objects in frames of CoTask 1 and CoTask 2.
Answer Format: A list of dictionaries with: `[{"head": str, "relation": str, "tail": str, "start_frame": int, "end_frame": int}, ...]`

CoTask 4: Temporal Relation Extraction

This task identifies temporal interactions (*e.g.*, `carry`, `follow`) between objects across a sequence of frames.

Question Format: Identify actions among entities using spatial and temporal cues from CoTask 1-3.
Answer Format: A list of dictionaries with: `[{"head": str, "relation": str, "tail": str, "start_frame": int, "end_frame": int}, ...]`

4.2.1 Object-level video reasoning task reconstruction

We reconstruct the VideoQA task to align video frames with object-level annotations. In this work, we focus on two representative VideoQA benchmarks: NeXT-QA and STAR. NeXT-QA is built upon the same raw video data as the VidOR dataset but lacks object-level annotations such as bounding boxes and spatial or temporal relations. To address this limitation, we merge the annotations from VidOR into NeXT-QA by matching `video_ids`. We then uniformly sample 64 frames per video, along with their corresponding annotations. The choice of 64 frames is empirically validated to be optimal for enabling LLaVA-Video to function effectively (see Table 6 and Table 5). In contrast, STAR is natively constructed as an object-centric VideoQA dataset. It includes the full set of original video frames (ranging from 0 to 92) and provides comprehensive object-level annotations, which we directly leverage for CoTask construction.

4.2.2 CoTasks construction

As shown in Figure 2, we construct four foundational CoTasks from the original object-level VideoQA tasks, following the definitions introduced in Section 4.1. In NeXT-QA, the high-level questions (*question_0 (Q0)*) are manually annotated by referring to the entire video rather than specific frames. As a result, we cannot directly ground the relevant video frames for *Q0* without leveraging pretrained VideoLLMs. To address this, we utilize the state-of-the-art VideoLLM, Qwen2.5-VL-72B, to ground relevant objects and frames for *Q0*, generating *answer_1 (A1)* for CoTask 1. The inputs for this step include the filtered video frames, *Q0*,

System:
You are a vision-language reasoning assistant. You will be shown a sequence of 64 video frames and a question referring to entities and their interactions. Your job is to match the context in the question to one or more of the ground-truth entities, and determine which frames show those entities co-occurring or interacting.

Task:
- Extract a list of entities (from the set above) that are mentioned or implied in the question (must be 1 or more).
- Identify the frame indices (timestamps) between 1 and 64 where those entities appear together.
- Return a concise answer as a single JSON object without any markdown code fences or backticks.
- The number of timestamps must be greater than 0 and less than 17.
- Include only the most relevant keyframes that are clearly related to the question.

Output format:
```
{"entities": [/* list of extracted entity names */], "timestamps": [/* list of frame numbers from 1 to 64, length 1-16 */]}
```

Example:
Question: why did the adult point to the fruits on the table?
Ground-truth entities: `['0_baby', '1_table', '2_fruits', '3_fruits', '4_adult']`

Answer:
```
{"entities": ['0_baby', '2_fruits', '4_adult'], "timestamps": [1, 2, 8, 10]}
```

Your Turn:
Question: {{YOUR_QUESTION_HERE}}
Ground-truth entities: {{Ground-truth entities}}
Answer:

Table 1: Prompt template for CoTask 1 data generation: Frame localization based on object co-occurrence.

and identified subject/object entities, along with the prompt template shown in Table 1. We used four A100 GPUs for a total of 10 hours for this process. In contrast, the *Q0* questions in STAR are constructed with reference to relevant frames and their corresponding object-level annotations. We extract the grounded objects and frames directly from the STAR dataset and convert them into the CoTask 1 format without requiring a VideoLLM. CoTask 2 is built upon *A1* and aims to generate *question_2 (Q2)* by matching objects and their bounding boxes across the localized frames. Its inputs are *A1* and object trajectories. CoTask 3 focuses on extracting spatial relations within individual localized frames. The inputs for this task include *A1*, *A2*, and spatial *relation_instances*. Finally, CoTask 4 targets temporal relation extraction across frames, using *A1*, *A2*, *A3*, and temporal *relation_instances* as input.

5 Experiments

In this section, we evaluate the proposed CoTasks (CoTasks-NeXT-QA and CoTasks-STAR) using recent VideoLLMs (*e.g.*, Qwen2.5-VL-3/7/72B and LLaVA-Video-7B) to assess their impact on inference-time performance (§5.3). We further conduct an ablation study to examine the difficulty of solving CoTasks with these models, and to analyze the contribution of each CoTask component during inference (§5.4). Specifically, we evaluate the contributions of CoTask 1-2 (frame localization and object tracking) for grounding, and CoTasks 3-4 (spatiotemporal relation extraction) for higher-level reasoning. In addition, we present qualitative visualizations of selected data samples to illustrate the quality of the constructed CoTasks (§5.5). The generated responses are evaluated using a GPT-4-based automatic scoring framework (Majumdar et al., 2024) (§5.1).

5.1 LLM-based Evaluator

We employ GPT-4 as an automatic evaluator to score the generated responses (Majumdar et al., 2024). Given the model outputs and corresponding ground-truth answers, along with the evaluation prompt template shown in Table 2, GPT-4 assigns a score from 1 to 5 based on their semantic alignment. The evaluation criteria are detailed in the prompt template.

5.2 Dataset

We construct two CoTask datasets based on existing object-centric VideoQA benchmarks: NeXT-QA and STAR, resulting in **CoTasks-NeXT-QA** and **CoTasks-STAR**, respectively. For **CoTasks-NeXT-QA**, we utilize a subset of the original NeXT-QA samples due to limited access to the raw videos from the VidOR dataset. As shown in Table 3, the original dataset contains 5,440 videos and 47,692 questions. After filtering for available videos, we retain 3,821 videos and decompose each question into four foundational CoTask types, resulting in 43,392 CoTask samples. The distribution of question types in the validation set is illustrated in Figure 3, with an outer ring showing subcategories and an inner ring grouping them into Causal, Temporal, and Descriptive types. For **CoTasks-STAR**, we utilize all 3,946 videos available in the original STAR dataset. Each question is similarly decomposed into four CoTask questions, resulting in 211,316 CoTask samples, as summarized in Table 4.

System:
You are an AI assistant who will help me to evaluate the response given the question and the correct answer. To mark a response, you should output a single integer between 1 and 5 (including 1, 5).
- 5 means that the response perfectly matches the answer.
- 1 means that the response is completely different from the answer.

Example 1:
Question: Is it overcast?
Answer: no
Response: yes
Your mark: 1

Example 2:
Question: Who is standing at the table?
Answer: woman
Response: Jessica
Your mark: 3

Example 3:
Question: Are there drapes to the right of the bed?
Answer: yes
Response: yes
Your mark: 5

Your Turn:
Question: {{question}}
Answer: {{answer}}
Response: {{prediction}}

Table 2: Prompt template for GPT-4-based evaluation of open-ended videoQA responses. Evaluators assign scores from 1–5 based on alignment with ground-truth and generated answers.

Dataset	Video	Train	Valid	Test	Total
Original	**5,440**	34,132	4,996	8,564	47,692
Filtered	3,821	9,188	1,660	-	10,848
CoTasks	3,821	**36,752**	**6,640**	**-**	**43,392**

Table 3: Statistics for the CoTasks-NeXT-QA dataset. The "Original" row shows the number of questions in the original NeXT-QA dataset. The "Filtered" row reflects the subset for which video content is available. Each filtered question is reformulated into four foundational CoTask instances, resulting in the final statistics shown in the "CoTasks" row.

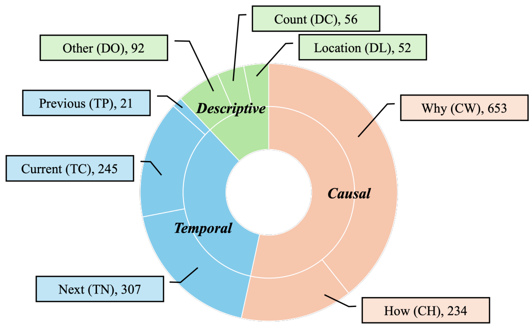

Figure 3: Distribution of question types in the CoTasks-NeXT-QA validation set. The inner ring groups questions into three high-level categories: Causal, Temporal, and Descriptive. The outer ring shows the corresponding fine-grained subcategories, illustrating the diversity of reasoning required.

5.3 Performance

We evaluate the effectiveness of CoTasks across state-of-the-art VideoLLMs using a GPT-4-based evaluator (see the prompt template in Table 13 in the appendix). As shown in Table 6, incorporating CoTasks consistently enhances model performance on CoTasks-NeXT-QA, with the most substantial improvement observed in the lightweight Qwen2.5-VL-3B model (+17.4 GPT-4 score). These results highlight a key limitation of current VideoLLMs in performing fine-grained, object-centric reasoning—particularly in smaller models. CoTasks help address this limitation by enriching inference with structured grounding and relational context.

Moreover, Table 7 shows that CoTasks significantly improve validation accuracy on the STAR dataset when using Qwen2.5-VL-3B, yielding a +34.3% gain over the baseline without CoTask prompting. Notably, this performance also surpasses that of the model fine-tuned on the STAR training set by +13.8%, demonstrating that structured grounding and relational context provided through CoTasks offer benefits beyond conventional instruction tuning with high-level question-answer pairs. We used one A100 GPU for completing this study. These findings suggest that CoTasks offer a promising approach for enhancing the reasoning capabilities of lightweight VideoLLMs, and they highlight a path toward developing fine-grained, object-centric VideoLLMs through structured multi-step prompting.

5.4 Ablation study

Per-CoTask Difficulty Analysis. Table 8 examines the difficulty of solving CoTasks using the best-performing model identified in Table 6. Specifically, we evaluate LLaVA-Video-7B on CoTasks 1–4 without fine-tuning, using the CoTasks-NeXT-QA validation set. All CoTasks yield GPT-4 evaluation scores below 30.0%, revealing substantial limitations of current VideoLLMs in addressing fundamental visual reasoning tasks. These include CoTask 1 (frame localization), CoTask 2 (object tracking), CoTask 3 (spatial relation extraction), and CoTask 4 (temporal relation extraction). The results underscore the complementary nature of the

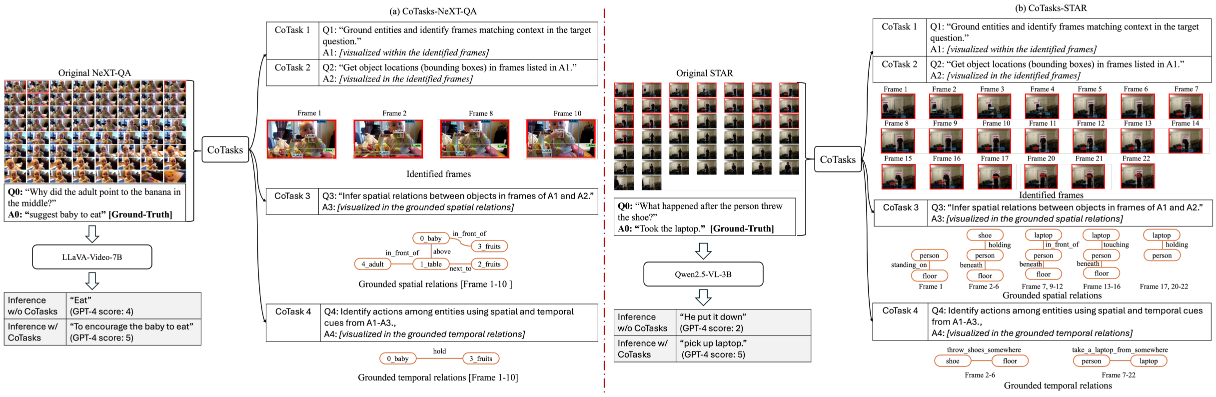

Figure 4: Qualitative case study on CoTasks-NeXT-QA (left) and CoTasks-STAR (right) comparing model inference with and without CoTasks. Highlighted video frames are selected via CoTask 1 (frame localization), with object tracking (CoTask 2) shown via bounding boxes. Grounded spatial and temporal relations from CoTasks 3 and 4 are visualized below. CoTasks significantly improve response quality, as reflected in GPT-4 evaluation scores.

Dataset	Video	Train	Valid	Total
Original	3,946	45,731	7,098	52,829
CoTasks	3,946	182,924	28,392	211,316

Table 4: Statistics for the CoTasks-STAR dataset. The dataset includes all videos and questions from the original STAR benchmark. Each original question is reformulated into four CoTask questions, resulting in a substantial increase in total training and validation samples.

CoTask design and highlight the need for targeted benchmarks to assess and improve foundational visual reasoning capabilities in VideoLLMs. We used two A100 GPUs for approximately five days to conduct the scores in Table 8 and Table 6. The prompt templates used for this analysis are provided in Tables 9, 10, 11, 12 in the appendix.

Impact of CoTask Subsets on Question Types. Table 5 presents an ablation study on LLaVA-Video-7B, evaluating the impact of different subsets of CoTasks on performance across various question types. Applying CoTasks 1–2 (frame localization and object tracking) improves average accuracy from 50.2 to 52.6, with notable gains in descriptive questions (e.g., +6.9 on DO and +7.3 on DL). CoTasks 3–4 (spatial and temporal relation extraction) also provide substantial improvements, especially for causal and descriptive categories, achieving the highest score on DC (count). When all CoTasks (1–4) are applied jointly, the model achieves the best overall performance (53.5 average), suggesting that each CoTask component contributes complementary information. These results highlight the future work on enhancing VideoLLMs' capabilities on structured, object-level reasoning tasks across diverse temporal and semantic categories. We used one A100 GPU for conducting this study.

5.5 Case study

Figure 4 presents a qualitative case study comparing inference results with and without CoTasks on two examples—one from CoTasks-NeXT-QA using LLaVA-Video-7B (left) and one from CoTasks-STAR using Qwen2.5-VL-3B (right). In each case, the top rows show the full video sequence with the frames selected by CoTask 1 (frame localization) highlighted in red. Below, CoTask 2 visualizes object tracking using bounding boxes over localized frames. CoTask 3 and CoTask 4 illustrate spatial and temporal relationships between entities, grounded from the detected object instances.

The examples demonstrate that incorporating CoTasks significantly improves the quality of video understanding. For instance, the inference for the NeXT-QA sample is refined from a vague "Eat" (score: 4) to a more specific and context-aware response: "To encourage the baby to eat" (score: 5). Likewise, in the STAR sample, the model's prediction improves from "He put it down" (score: 2) to a precise action description: "Pick up laptop." (score: 5). These results highlight how CoTasks contribute to grounding visual context and enhancing inference accuracy.

6 Conclusions

We present CoTasks, a structured framework for instruction tuning that enables VideoLLMs to per-

Model	CoTasks	Causal		Temporal			Descriptive			Avg
		CW	CH	TP	TC	TN	DC	DL	DO	
LLaVA-video-7B	–	51.3	54.8	28.6	48.0	36.4	65.2	63.0	72.3	50.2
	1–2	53.4	**55.9**	**32.1**	49.4	38.0	70.5	**72.1**	**78.5**	52.6
	3–4	52.6	55.5	**32.1**	51.4	36.6	**73.2**	63.0	73.9	51.8
	1–4	**55.1**	54.8	27.4	**51.9**	**39.3**	70.5	68.8	76.9	**53.5**

Table 5: Ablation study on LLaVA-Video-7B across question types. CoTasks 1–2 (frame localization + tracking) and 3–4 (spatial/temporal relation extraction) yield distinct gains, while combining all tasks (1–4) leads to the best average performance. CW = Why, CH = How, TP = Previous, TC = Current, TN = Next, DC = Count, DL = Location, DO = Other.

Model	CT	C.	T.	D.	Avg.
Qwen2.5-VL-3B	–	35.0	21.6	13.8	27.8
	✓	**49.6**	**32.5**	**61.9**	**45.2**
Qwen2.5-VL-7B	–	52.1	39.0	66.1	49.3
	✓	**55.3**	**39.8**	66.1	**51.3**
Qwen2.5-VL-72B	–	52.7	38.9	67.6	49.7
	✓	**55.3**	**41.3**	**70.0**	**52.3**
LLaVA-video-7B	–	52.2	41.1	67.9	50.2
	✓	**55.0**	**44.3**	**73.0**	**53.5**

Table 6: GPT-4 evaluation scores with (✓) and without (–) CoTasks-NeXT-QA. CT = CoTasks(1–4), C. = Causal, T. = Temporal, D. = Descriptive, Avg. = Average. **CoTasks consistently enhance performance across all models, yielding a notable +17.4 point gain for Qwen2.5-VL-3B.**

Model	Dataset	PT	FT	Acc.(%)
Qwen2.5-VL-3B	STAR	–	–	31.1
		–	✓	51.6 (+20.5)
		✓	–	**65.4 (+34.3)**

Table 7: Accuracy on the STAR dataset using Qwen2.5-VL-3B under different combinations of CoTask-style prompting (PT) and fine-tuning (FT). Prompting alone significantly boosts performance, with gains of +34.3 points.

CoTask	C.	T.	D.	Avg.
CoTask 1	17.3	15.3	26.9	17.7
CoTask 2	0.7	0.6	4.0	1.1
CoTask 3	23.3	25.1	25.8	24.2
CoTask 4	9.9	11.2	13.6	10.8

Table 8: Validation accuracy (%) of **LLaVA-Video-7B** (64-frame input, no fine-tuning) on CoTasks 1–4, evaluated under GPT-4 rubric types: **C.** (causal), **T.** (temporal), and **D.** (descriptive).

form chain-of-thought (CoT) reasoning grounded in fine-grained object-level video understanding. By decomposing complex video questions into four foundational sub-tasks—frame localization, entity tracking, spatial relation extraction, and temporal relation extraction—CoTasks provide explicit, interpretable supervision for spatiotemporal reasoning. Experiments on the NeXT-QA and STAR benchmarks demonstrate that CoTasks substantially improve inference performance, particularly for lightweight models such as Qwen2.5-VL-3B, which gains +17.4 GPT-4 points overall. The improvements span causal, temporal, and descriptive reasoning types, confirming the benefit of compositional task design. Our findings highlight the importance of structured, intermediate supervision for advancing compositional video understanding. Future work includes fine-tuning VideoLLMs directly on CoTasks and extending the framework to multi-modal and open-domain video reasoning settings.

Limitations

Our approach presents several limitations. First, CoTasks require object-level annotations to expand existing VideoQA tasks, which may limit applicability in datasets lacking such annotations. Second, since each CoTask builds upon the preceding one, the quality of CoTask 1 (frame localization) directly affects the construction and reliability of subsequent CoTasks. Third, we evaluate model performance by prompting recent VideoLLMs with CoTasks rather than fine-tuning the models on the CoTask formulations. We leave fine-tuning with CoTasks as an important direction for future work.

References

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. arXiv preprint arXiv:2308.12966.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. 2025. Qwen2.5-vl technical report. arXiv preprint arXiv:2502.13923.

Joao Carreira, Eric Noland, Chloe Hillier, and Andrew Zisserman. 2019. A short note on the kinetics-700 human action dataset. arXiv preprint arXiv:1907.06987.

Zhenfang Chen et al. 2024a. Enhancing video-llm reasoning via agent-of-thoughts distillation. arXiv preprint arXiv:2412.01694.

Zhenfang Chen et al. 2024b. Understanding long videos with multimodal language models. arXiv preprint arXiv:2403.16998.

Google DeepMind. 2023. Gemini: Google's multimodal ai models. https://deepmind.google/technologies/gemini.

Haotian Liu, Xiang Lin, Chunyuan Li, Shiqi Chen, Kevin Lin, Yuheng Xu, Zihan Wang, Jianwei Zhang, and Chunyuan Zhang. 2024a. Llava-o1: Let vision language models reason step-by-step. arXiv preprint arXiv:2403.08195.

Haotian Liu, Chunyuan Zhang, Yuheng Xu, Zihan Wang, Jianwei Zhang, et al. 2023. Llava: Large language and vision assistant. arXiv preprint arXiv:2304.08485.

Peiqi Liu, Zhanqiu Guo, Mohit Warke, Soumith Chintala, Nur Muhammad Mahi Shafiullah, and Lerrel Pinto. 2024b. Dynamem: Online dynamic spatio-semantic memory for open world mobile manipulation. arXiv preprint arXiv:2411.04999.

Peiqi Liu, Yaswanth Orru, Chris Paxton, Nur Muhammad Mahi Shafiullah, and Lerrel Pinto. 2024c. Ok-robot: What really matters in integrating open-knowledge models for robotics. arXiv preprint arXiv:2401.12202.

Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. 2024. Video-chatgpt: Towards detailed video understanding via large vision and language models. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL 2024).

Arjun Majumdar, Anurag Ajay, Xiaohan Zhang, Pranav Putta, Sriram Yenamandra, Mikael Henaff, Sneha Silwal, Paul Mcvay, Oleksandr Maksymets, Sergio Arnaud, Karmesh Yadav, Qiyang Li, Ben Newman, Mohit Sharma, Vincent Berges, Shiqi Zhang, Pulkit Agrawal, Yonatan Bisk, Dhruv Batra, Mrinal Kalakrishnan, Franziska Meier, Chris Paxton, Sasha Sax, and Aravind Rajeswaran. 2024. Openeqa: Embodied question answering in the era of foundation models. In Conference on Computer Vision and Pattern Recognition (CVPR).

meta. 2024. Llama 3. Accessed: 2025-02-04.

Ming Nie, Dan Ding, Chunwei Wang, Yuanfan Guo, Jianhua Han, Hang Xu, and Li Zhang. 2024. Slowfocus: Enhancing fine-grained temporal understanding in video llm. In Proceedings of the Conference on Neural Information Processing Systems (NeurIPS).

OpenAI. 2023. Gpt-4 with vision. https://openai.com/research/gpt-4v-system-card.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. Proceedings of the International Conference on Machine Learning (ICML).

Xindi Shang, Donglin Di, Junbin Xiao, Yu Cao, Xun Yang, and Tat-Seng Chua. 2019. Annotating objects and relations in user-generated videos. pages 279–287.

Hao Shao, Shengju Qian, Han Xiao, Guanglu Song, Zhuofan Zong, Letian Wang, Yu Liu, and Hongsheng Li. 2024. Visual cot: Advancing multi-modal language models with a comprehensive dataset and benchmark for chain-of-thought reasoning. In Proceedings of the Conference on Neural Information Processing Systems (NeurIPS).

Michael Tschannen et al. 2023. Siglip: Scaling up image-language pretraining with contrastive mixture of experts. arXiv preprint arXiv:2303.15343.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. 2024. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. arXiv preprint arXiv:2409.12191.

Xiao Wang, Qingyi Si, Jianlong Wu, Shiyu Zhu, Li Cao, and Liqiang Nie. 2025. Adaretake: Adaptive redundancy reduction to perceive longer for video-language understanding. arXiv preprint arXiv:2503.12559.

Jason Wei et al. 2022. Chain of thought prompting elicits reasoning in large language models. arXiv preprint arXiv:2201.11903.

Bo Wu, Shoubin Yu, Zhenfang Chen, Joshua B Tenenbaum, and Chuang Gan. 2021. Star: A benchmark for situated reasoning in real-world videos. In Advances in Neural Information Processing Systems.

Yitong Xiao, Zhe Gan, Yu Cheng, et al. 2021. Next-qa: Next phase of question answering to benchmark video reasoning. In CVPR.

Meishan Zhang et al. 2023. Video-of-thought: A dynamic scene graph approach for video reasoning. arXiv preprint arXiv:2309.15402.

Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. 2024. Video instruction tuning with synthetic data.

Orr Zohar, Xiaohan Wang, Yonatan Bitton, Idan Szpektor, and Serena Yeung-Levy. 2025. Video-STar: Self-training enables video instruction tuning with any supervision. In The Thirteenth International Conference on Learning Representations.

A Appendix

A.1 Answers to potential questions.

Q1: Why were these 4 tasks selected for the CoTasks pipeline? A1: The combination of object-level tasks (object identification and tracking) with higher-level reasoning tasks (spatial relations and actions) creates a bridge for understanding fine-grained interactions in video, enabling more structured and compositional reasoning.

Q2: Is the pipeline for constructing CoTasks the same across different datasets (e.g., NeXT-QA, STAR)? A2: No, the pipeline differs for each dataset. For NeXT-QA, object-level information was retrieved from the VIDOR dataset annotations and aligned with the NeXT-QA questions and answers. In contrast, the STAR dataset is better structured and allows construction based on a combination of predefined catalogs, including objects, persons, relations, and object bounding boxes.

Q3: How is video selection and reduction to 64 frames performed? A3: A uniform sampling strategy is used to select 64 frames from the full set of video frames. These selected frames are then re-indexed to align with annotations, which originally refer to the full sequence of frames.

Q4: How do LLMs support the implementation of CoTasks? A4: Unlike exact textual matching, LLMs enable semantic understanding of questions and their referenced entities, allowing better alignment with ground-truth annotations for CoTasks construction.

Q5: Do you plan to open-source CoTasks? A5: Yes, we plan to open-source CoTasks after the review process.

Q6: Why don't you use multiple-choice style datasets? A6: Because the multiple-choice datasets provide all answer options as part of the input and require the model to select the correct one. This setup limits the model's ability to generate free-form, compositional reasoning, which is essential for evaluating fine-grained understanding.

Q7: Is building the dataset that includes CoTasks the main contribution? Don't existing datasets already include tasks like 'where' or 'who'? A7: Yes, the main contribution is the construction of the CoTasks dataset, which decomposes complex video reasoning questions into four structured subtasks. Existing datasets like NeXT-QA and STAR do not provide such decomposed, chain-of-thought supervision involving explicit "where" or "who" reasoning steps.

Q8: How were the CoTasks annotated? Were human annotators involved or were external models used? A8: The CoTasks were generated automatically using the object-level annotations and question structures from existing datasets. There is no indication that manual annotation was involved; instead, structured templates and model-based grounding (*e.g.*, with Qwen2.5-VL) were used.

Q9: The improvement for 3B models is large. Why is the gain smaller for larger models? A9: Smaller models like 3B benefit more from structured intermediate supervision, as they lack deep pretraining knowledge. Larger models already possess stronger reasoning ability, so CoTasks yield less marginal improvement due to diminishing returns.

Q10: Is the order of CoTask 1–4 important? A10: Yes, the order reflects a logical reasoning flow: starting from frame grounding (CoTask 1), through entity tracking (CoTask 2), to spatial (3) and temporal (4) relation reasoning. This sequence aligns with how humans and models naturally build understanding step-by-step.

Q11: Why was this submitted to an NLP conference instead of a CV or ML venue? A11: Although the tasks involve video, the focus is on chain-of-thought reasoning using instruction tuning—a core NLP methodology. CoTasks are embedded as textual prompts for LLMs, making NLP conferences such as ARR/ACL the most relevant venue.

Q12: Are the CoT examples constructed from a dataset independent of the evaluation set? A12: Yes, CoTasks are created based on training data from existing datasets like NeXT-QA and STAR. The evaluation uses separate validation/test splits, ensuring there is no data leakage between generation and evaluation.

Q13: Can future models learn to decide the task sequence and content themselves? A13: Yes, the paper suggests this as a promising extension. Enabling models to autonomously plan and select intermediate reasoning steps (i.e., dynamic chain-of-task inference) would make the pipeline more adaptive and closer to real-world reasoning.

A.2 Prompt template used in the evaluation.

We provide all prompt templates used in Experiments (see §5).

Task:
You are a vision-language reasoning model. Your goal is to extract relevant entities from a given question and identify the video frames (1–64) where those entities co-occur.

Instructions:
- First, read the question carefully and determine which entities (from the grounded set) are directly mentioned or implied.
- Then, find the frames (from 1 to 64) where those entities appear together.
- Output your answer as a JSON object with two keys:
  - `"entities"`: list of relevant entity names (e.g., `"0_adult"`, `"3_handbag"`)
  - `"timestamps"`: list of frame numbers where those entities are present together
- The number of timestamps must be between 1 and 16.
- Do not include any extra explanation, markdown, or formatting—just return a valid JSON object.

Example Input:
Q: Ground entities and identify frames matching context in the target question.
Contextual question: "What else does the man in yellow carry aside from a black laptop bag?"

Output format:
```
{"entities": ["0_adult",
"3_handbag"], "timestamps": [1, 5,
9, 12, 15]}
```

Your Turn:
Q: Ground entities and identify frames matching context in the target question.
Contextual question: "What else does the man in yellow carry aside from a black laptop bag?"
Res:

Table 9: Prompt for CoTask 1: Entity grounding and timestamp prediction based on video QA context. The result shown in Table 8.

Task:
You are a visual perception assistant. Based on a contextual question and prior grounding results, your task is to identify the bounding boxes of relevant entities in selected video frames.

Contextual question (Q0):
What else does the man in yellow carry aside from a black laptop bag?

Reasoning question (Q2):
Get object locations (bounding boxes) in frames listed in A1.

Grounded input (A1):
```
{"entities": ["0_adult",
"3_handbag"], "timestamps": [1, 5,
9, 12, 15]}
```

Instructions:
- For each frame listed in `"timestamps"`, detect the presence of the listed `"entities"`.
- For each detected entity in a frame, return:
  - `"label"`: the entity ID (e.g., `"0_adult"`, `"3_handbag"`)
  - `"bbox"`: bounding box in the format `[x1, y1, x2, y2]`
- Output your result as a JSON list of dictionaries, each with:
  - `"frame"`: the frame number
  - `"objects"`: list of detected objects and their bounding boxes
- Do not include explanations, markdown, or extra text—only return valid JSON.

Output format example:
```
[{"frame": 1, "objects": ["label":
"0_adult", "bbox": [262, 2, 400,
333], "label": "3_handbag", "bbox":
[294, 48, 393, 146]]},...]
```

Your Turn:
Contextual question: What else does the man in yellow carry aside from a black laptop bag?
Reasoning question: Get object locations (bounding boxes) in frames listed in A1.
Entities: `["0_adult", "3_handbag"]`
Frames: `[1, 5, 9, 12, 15]`
Res:

Table 10: Prompt for CoTask 2: Predicting bounding boxes for grounded entities across localized frames. The result shown in Table 8.

Task:
You are a spatial reasoning assistant. Based on the contextual question and prior grounding information, your task is to infer spatial relationships between visual entities in specific video frames.

Contextual question (Q0):
What else does the man in yellow carry aside from a black laptop bag?

Reasoning question (Q3):
Infer spatial relations between objects in frames of A1 and A2.

Supporting input:
A1 (Entities and timestamps):
```
{"entities": ["0_adult",
"3_handbag"], "timestamps": [1, 5,
9, 12, 15]}
```
A2 (Object bounding boxes per frame):
```
[{"frame": 1, "objects": [{"label":
"0_adult", "bbox": [262, 2, 400,
333]}, {"label": "3_handbag",
"bbox": [294, 48, 393, 146]}]},
{"frame": 5, "objects": [{"label":
"0_adult", "bbox": [355, 17, 520,
273]}, {"label": "3_handbag", "bbox":
[386, 0, 495, 87]}]}, {"frame": 9,
"objects": [{"label": "0_adult",
"bbox": [369, 12, 480, 188]}]},
{"frame": 12, "objects": [{"label":
"0_adult", "bbox": [331, 14, 421,
140]}]}, {"frame": 15, "objects":
[]}]}
```

Instructions:
- For each frame, determine if two entities are spatially related (e.g., "next_to", "behind", "on", etc.).
- A valid spatial relation must occur in individual frames.
- For each detected spatial relationship, return:
  - "head": the source entity
  - "relation": the spatial relationship
  - "tail": the target entity
  - "start_frame": the first frame where the relation is observed
  - "end_frame": the last frame where the relation holds
- Output your result as a JSON list of dictionaries.
- Do not include explanations, markdown, or extra text—only return valid JSON.

Output format example:
```
[ {"head": "0_adult", "relation":
"next_to", "tail": "3_handbag",
"start_frame": 1, "end_frame": 5},
... ]
```

Your Turn:
Contextual question: What else does the man in yellow carry aside from a black laptop bag?
Reasoning question: Infer spatial relations between objects in frames of A1 and A2.
Entities: ["0_adult", "3_handbag"]
Frames: [1, 5, 9, 12, 15]
Bounding boxes: see A2 above
Res:

Table 11: Prompt for CoTask 3: Inferring spatial relationships between grounded objects across localized frames. The result shown in Table 8.

Task:
You are a visual reasoning agent. Your job is to analyze spatial and temporal cues from a sequence of video frames to infer action relationships between entities.

Contextual question (Q0):
What else does the man in yellow carry aside from a black laptop bag?

Reasoning question (Q4):
Identify actions among entities using spatial and temporal cues from A1–A3.

Supporting input:
A1 (Entities and timestamps):
```
{"entities": ["0_adult",
"3_handbag"], "timestamps": [1, 5,
9, 12, 15]}
```
A2 (Bounding boxes per frame):
```
[{"frame": 1, "objects": [{"label":
"0_adult", "bbox": [262, 2, 400,
333]}, {"label": "3_handbag",
"bbox": [294, 48, 393, 146]}]},
{"frame": 5, "objects": [{"label":
"0_adult", "bbox": [355, 17, 520,
273]}, {"label": "3_handbag", "bbox":
[386, 0, 495, 87]}]}, {"frame": 9,
"objects": [...]}, {"frame": 12,
"objects": [...]}, {"frame": 15,
"objects": []}]
```
A3 (Spatial relations):
```
[{"head": "0_adult", "relation":
"next_to", "tail": "3_handbag",
"start_frame": 1, "end_frame": 12}]
```

Instructions:
- Infer **actions** between entities (e.g., "carry", "hold", "push", "pull") using:
  - Proximity and overlap in bounding boxes (A2)
  - Persistent spatial relations (A3)
- For each inferred action, return a dictionary with:
  - "head": the acting entity
  - "relation": the action verb
  - "tail": the affected entity
  - "start_frame": first frame of the action
  - "end_frame": last frame of the action
- Return a list of such dictionaries in valid JSON format.
- Do not include explanations, markdown, or commentary—only the JSON.

Output format example:
```
[{"head": "0_adult", "relation":
"carry", "tail": "3_handbag",
"start_frame": 1, "end_frame": 12}]
```

Your Turn:
Contextual question: What else does the man in yellow carry aside from a black laptop bag?
Reasoning question: Identify actions among entities using spatial and temporal cues from A1–A3.
Entities: ["0_adult", "3_handbag"]
Frames: [1, 5, 9, 12, 15]
Bounding boxes and spatial relations: see A2 and A3 above
Res:

Table 12: Prompt for CoTask 4: Temporal action reasoning based on bounding box and spatial relation history. The result shown in Table 8.

Task:
You are a visual reasoning assistant. Given a series of video frames and visual annotations, your goal is to answer a high-level question (Q0) about an event involving specific entities. You will be provided supporting information from auxiliary visual sub-tasks (Q1–Q4), which ground entities, detect object locations, infer spatial relationships, and determine actions.

Your response must be a concise phrase that best answers Q0 using reasoning based on the visual and relational evidence provided.

Instructions:
- Use entity co-occurrence (A1), object bounding boxes (A2), spatial relations (A3), and inferred actions (A4) to support your answer.
- Base your answer on what the visual evidence consistently supports across the relevant frames.
- Respond with a **short phrase** (e.g., an object or action) directly answering Q0.

Input:
Q0: what else does the man in yellow carry aside from a black laptop bag?
A1: `{'entities': ['0_adult', '3_handbag'], 'timestamps': [1, 5, 9, 12, 15]}`
A2: `[{'frame': 1, 'objects': [{'label': '0_adult', 'bbox': [262, 2, 400, 333]}, {'label': '3_handbag', 'bbox': [294, 48, 393, 146]}}, ...}]`
A3: `[{'head': '0_adult', 'relation': 'next_to', 'tail': '4_handbag', 'start_frame': 1, 'end_frame': 12}, ...]`
A4: `[{'head': '0_adult', 'relation': 'carry', 'tail': '4_handbag', 'start_frame': 1, 'end_frame': 12}, ...]`

Output format:
Respond with a short phrase that answers Q0 using the evidence from A1–A4.
Respond: book

Table 13: Prompt template for evaluating the original task (the result shown in Table 6).

Responsible NLP Research Checklist

A. For every submission

- A1. Did you discuss the limitations of your work? **Yes, see section Limitations**
- A2. Did you discuss any potential risks of your work? **No, our proposed CoTasks was constructed existing annotated object-level videoQA tasks NeXT-QA and STAR.**
- A3. Do the abstract and introduction summarize the paper's main claims? **Yes.**

B. Did you use or create scientific artifacts?

- B1. Did you cite the creators of artifacts you used? **Yes.**
- B2. Did you discuss the license or terms for use and/or distribution of any artifacts? **Yes.**
- B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For artifacts you create, do you specify intended use and whether that is compatible with the original access conditions? **Yes.**
- B4. Did you discuss the steps taken to check whether the data that was collected/used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it? **Yes.**
- B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, linguistic phenomena, demographic groups represented, etc.? **Yes.**
- B6. Did you report relevant statistics like the number of examples, details of train/test/dev splits, etc. for the data that you used/created? **Yes.**

C. Did you run computational experiments?

- C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used? **Yes. We report the number of parameters in the models used and GPU hours for conducting all experiments.**
- C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values? **Yes.**
- C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc., or just a single run? **Yes.**

D. Did you run human-subjects experiments?

- D1. Did you provide details on the task setup, instructions, and payments? **No, our proposed Cotasks does not include any human-subject experiment.**
- D2. Did you describe any harm mitigation strategies taken during data collection? **No, our proposed Cotasks does not include any human-subject experiment.**
- D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? **No, our proposed Cotasks does not include any human-subject experiment.**
- D4. Was the data collection protocol approved (or determined exempt) by an ethics review board? **No, our proposed Cotasks does not include any human-subject experiment.**
- D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data? **No, our proposed Cotasks does not include any human-subject experiment.**

E. Did you use AI assistants (e.g., ChatGPT, Copilot) in your research, coding, or writing?

- E1. If you used any AI assistants, did you include information about your use? **Yes, we used GPT-4 as an LLM evaluator to assess how closely the generated answer matches the ground truth. In addition, we used ChatGPT to correct the English grammar in our paper and to generate code for data preprocessing.**