Qeios

Research Article

VideoLLM Knows When to Speak: Enhancing Time-Sensitive Video Comprehension with Video-Text Duet Interaction Format

Yueqian Wang[1], Xiaojun Meng[2], Yuxuan Wang[3], Jianxin Liang[1], Jiansheng Wei[2], Huishuai Zhang[1], Dongyan Zhao[1,4]

1. Wangxuan Institute of Computer Technology, Peking University, China; 2. Huawei Noah's Ark Lab, China; 3. Beijing Institute for General Artificial Intelligence, China; 4. National Key Laboratory of General Artificial Intelligence, China

Recent researches on video large language models (VideoLLM) predominantly focus on model architectures and training datasets, leaving the interaction format between the user and the model under-explored. In existing works, users often interact with VideoLLMs by using the entire video and a query as input, after which the model generates a response. This interaction format constrains the application of VideoLLMs in scenarios such as live-streaming comprehension where videos do not end and responses are required in a real-time manner, and also results in unsatisfactory performance on time-sensitive tasks that requires localizing video segments. In this paper, we focus on a video-text duet interaction format. This interaction format is characterized by the continuous playback of the video, and both the user and the model can insert their text messages at any position during the video playback. When a text message ends, the video continues to play, akin to the alternative of two performers in a duet. We construct MMDuetIT, a video-text training dataset designed to adapt VideoLLMs to video-text duet interaction format. We also introduce the Multi-Answer Grounded Video Question Answering (MAGQA) task to benchmark the real-time response ability of VideoLLMs. Trained on MMDuetIT, MMDuet demonstrates that adopting the video-text duet interaction format enables the model to achieve significant improvements in various time-sensitive tasks (76% CIDEr on YouCook2 dense video captioning, 90% mAP on QVHighlights highlight detection and 25% R@0.5 on Charades-STA temporal video grounding) with minimal training efforts, and also enable VideoLLMs to reply in a real-time manner as the video plays. Code, data and demo are available at: https://github.com/yellow-binary-tree/MMDuet.

Corresponding author: Dongyan Zhao, zhaodongyan@pku.edu.cn

1. Introduction
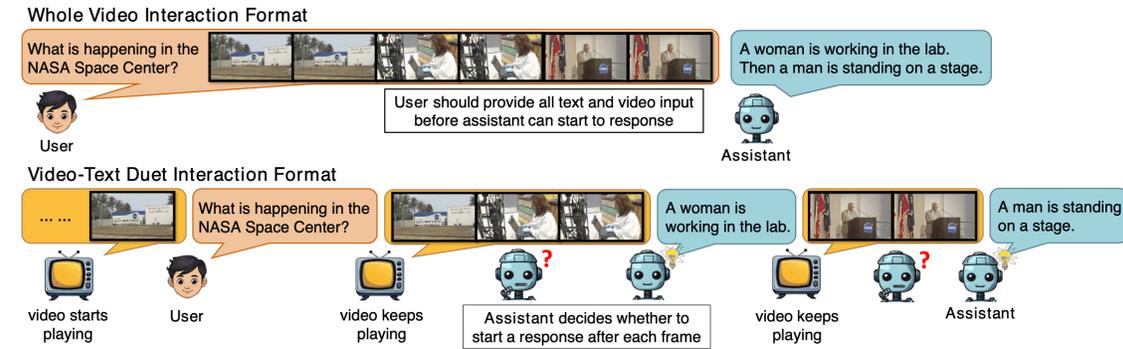
Figure 1. An example of the common Whole Video Interaction Format and our Video-Text Duet Interaction Format.

Videos are becoming an increasingly important medium to acquire information on a daily basis. Powered by recent advancements in large language models (LLMs) [1][2][3][4][5] and vision encoders [6][7][8][9][10], several video large language models (VideoLLM) [11][12][13][14][15][16] have already demonstrated strong abilities for holding conversations and answering questions about videos. A common feature of these models is using visual encoders to encode all frames sampled from the entire video at first, and integrate them into text input by concatenating them to input embeddings or using cross attention.

Recent research on VideoLLMs has primarily concentrated on model architectures and training datasets, with limited exploration of the interaction format between the user and the model. In this paper, the "interaction format" of VideoLLMs comprises the following two aspects: (1) a chat template used to convert input sources, e.g., video, user text query, and model response, into a sequence of tokens; (2) a turn-taking rule organizing inputs of different sources to finalize an interaction format. For example, for most existing VideoLLMs, the interaction format is: (1) for the chat template, the model uses (frames sampled from) the full video and a text query as input, and then outputs a response; (2) for the turn-taking rule, usually the model is permitted to take its turn to generate a response when both the whole video content and user query have ended, e.g., when an

<eos> token is explicitly provided. We refer to this traditional interaction method as "*whole video*" in the rest of this paper.

However, this all-along used *whole video* interaction has the following three defects, which hinder the performance and real-world usage scenarios of VideoLLMs: Firstly, it does not admit timely interactions. As the video is often input as a whole, this limits its usage in more scenarios like live broadcasts or surveillance videos, in which the video does not end at a specific time. Even if we can segment the video into multiple fixed-length clips for input, the model still cannot generate responses in a real-time manner when necessary. Instead, it must wait until the end of the clip before starting to reply, which negatively impacts the timeliness of the interaction. Secondly, it performs unfavorably on time-sensitive video comprehension tasks. In this paper we use **"time-sensitive tasks"** to refer to tasks in which the model is required to provide responses that include specific times in the video, such as temporal video grounding[17][18][19], video highlight detection[20], dense video captioning[21][17], grounded video question answering[22], etc. Existing approaches refer to the video timeline by generating text that contains time information, such as second numbers like "3.5 − 7.1 seconds" or timeline percentages like "from 30 to 50". However, the performance of these models has been consistently unsatisfactory, possibly due to LLMs' limited ability to handle numbers[23]. Thirdly, it suffers from sub-optimal performance of describing long videos. When generating responses relevant to a specific segment in a lengthy video, the model needs to first implicitly "retrieve" this relevant segment from the entire video that is already encoded in the context, which is challenging for most VideoLLMs[24].

In this work, we formalize the Video-Text Duet Interaction Format, an interaction method that aims to enhance VideoLLMs by addressing the aforementioned issues. An illustration of the *whole video* interaction format and the video-text duet interaction format is shown in Fig. 1. With our video-text duet interaction format, the video is continuously played and input to the model frame-by-frame. Both the user and model can insert their text messages right after any frame during the video play. When a text message ends, the video continues to play, akin to the show of two performers in a duet. This is implemented by also defining the "video stream" as a participant in the dialogue, with its messages only comprising video frames. When a dialogue turn from either the user or the model ends, the video stream can have the floor and input video frames to the model until another turn is started by either the user or the model. This improves the timeliness of interaction and better suits real-world applications such as live-streaming or surveillance video comprehension. Moreover, by inserting

responses to the video where is most relevant, the model can learn to generate responses by referencing a smaller but fine-grained fraction of the video before this position. In this manner, it facilitates information retrieval to describe lengthy videos, as well as enables a response to be "grounded" at the targeted position of the video. We believe this design contributes to addressing the above discussed three issues of existing VideoLLMs.

To prove the effectiveness of the video-text duet interaction format, we construct MMDuetIT, a dataset to facilitate the training of a versatile VideoLLM following the video-text duet interaction format. MMDuetIT is constructed by reformatting existing dense video captioning and temporal video grounding datasets by inserting each turn of the conversation at a proper position within the video. We also propose Multi-Answer Video Grounded QA (MAGQA), a novel task that requires the model to generate answers at appropriate positions within the video in a real-time manner, to align with potential applications of live-streaming video comprehension. We also train MMDuet, a VideoLLM that implements our proposed video-text duet interaction format. Initialized with LLaVA-OneVision[14] and trained with MMDuetIT at a low cost, MMDuet achieves significant performance improvement in various time-sensitive tasks, and is able to generate responses in real-time as the video plays.

2. Related Works

2.1. Video Large Language Models

The advancement of large language models (LLMs) and visual encoders has led to numerous efforts on their integration, aiming to utilize the powerful understanding and generation abilities of existing LLMs for video-related tasks[11][12][13][14][16][25]. These models exhibit a decent ability of video understanding such as captioning or summarizing[25]. However, their performance on time-sensitive tasks is still unsatisfactory, primarily due to two reasons: First, many models are trained on datasets in which video contents are relatively static with annotations not dense enough[15]. It thus constrains the model to discern the temporal differences across various parts of videos. Second, differing from image models that can easily capture word patch alignment[26], most VideoLLMs are not explicitly trained to distinguish differences between different clips in the same video.

Recent works attempt to empower VideoLLMs with the ability to localize and represent segments in videos, and thus achieve better performance on tasks like temporal video grounding or dense video captioning. These works explore new ways on how to easily represent video clips with texts, such as second numbers of timestamp (TimeChat[27]), timeline percentage (VTimeLLM[28]) or using special textual tokens (VTG-LLM[29], Grounded-VideoLLM[30]). However, their performance has not been satisfactory yet, possibly due to LLMs' limited ability to accurately count and generate numbers[23] to localize each video frame. To alleviate this issue, HawkEye[31] uses a coarse-grained method by referring to a larger fraction of the video, but it requires multiple rounds of recursive grounding to precisely locate a segment and may not express multiple segments at a time.

The work most similar to our motivation is VideoLLM-Online[32], which proposes a framework named LIVE for training VideoLLMs to interrupt video streams and insert responses. However, they only finetune a model on Ego4D[33] and COIN[34] to demonstrate the LIVE training and inference, and do not explore on how the model capabilities vary with this new type of interaction, especially the zero-shot performance on time-sensitive tasks.

Our work differs from VideoLLM-Online at: Firstly, providing a more general description of the video-text dual interaction format, including a wider variety of criteria for determining whether a response should be generated, and its application on new tasks such as temporal video grounding and grounded question answering; Secondly, introducing a new dataset MMDuetIT and the method on building such datasets; Thirdly, proposing a new task MAGQA; Lastly, proposing a more powerful model MMDuet that has state-of-the-art performance on various time-sensitive tasks and zero-shot generalization ability.

3. The Video-Text Duet Interaction Format

In Sec. 1, we have defined the concept of "interaction format" with two aspects (i.e., chat template & turn-taking rule), as well as the drawbacks of the commonly-used *whole video* interaction format. Now we re-emphasize and formalize our video-text duet interaction format, which is completely different from previous to implement VideoLLMs.

1. For the chat template, inspired by but different from the LIVE framework which is used to implement VideoLLM-Online[32], we consider the video stream as a conversation participant just

like the role of user/assistant, and the input sequence consists of alternating turns among these three roles. Take the LLM Qwen2[5] as an example, the token sequence input to the model should be as follows

---

`<im_start>`system\n[system-prompt]`<im_end>`
`<im_start>`stream\n`<frame>`···`<frame>``<im_end>`
`<im_start>`user\n[user-message]`<im_end>`
`<im_start>`stream\n`<frame>`···`<frame>``<im_end>`
`<im_start>`assistant\n[assistant-message]`<im_end>`
`<im_start>`stream\n`<frame>`···

---

2. For the turn-taking rule, when the turn of the user or assistant ends, the video stream can take the floor and start its turn to input video frames. When each single frame is consumed, both the user and the assistant role can interrupt the video stream at any time, and start its own turn to query or generate a response, as totally decided by the user or the assistant, respectively.

4. MMDuet: Our Proposed VideoLLM

4.1. Model Structure

We propose MMDuet, a model trained following the video-text duet interaction format, which can thus autonomously decide at what position in the video to generate what response. Like almost all existing VideoLLMs, MMDuet consists of three components: 1) a visual encoder that encodes sampled frames from the video to visual feature, 2) a projector that transforms the encoded visual feature to a list of visual tokens that is aligned into the LLM textual embedding space, and 3) a transformer-decoder-based LLM that takes both textual and visual tokens as input and uses its language modeling head to predict the next token. Note that we simply use a linear layer instead of video-level or sliding-window Qformers[11][27] as the projector, as in the video-text duet interaction format each frame is added to LLM input sequence independently.

The only difference in model structure between our MMDuet and existing VideoLLMs is that we add two more heads in addition to the language modeling head (LM Head) of the LLM, namely the **informative head** and the **relevance head**, for determining whether to start a response after each

frame. Each head is a linear layer and has a weight with shape $h \times 2$, where h is the hidden size of the used LLM. Each head takes the final layer hidden state of the last visual token of each frame as input, and performs a binary classification. To be specific, 1) the informative head is designed to predict how much new information is acquired upon viewing the current frame. If the model can obtain a "significant amount" of new information upon viewing a new frame (which we will further discuss in Sec. 5.1), it should classify this frame as *TRUE* category; otherwise, it should classify it as *FALSE*. 2) The relevance head is designed to predict whether the current frame is related to the user query. Similarly, *TRUE* category means to be related, while *FALSE* means not. We denote the probability of *TRUE* category of informative head and relevance head as **informative score** and **relevance score** for each sampled video frame. These two scores will be used to decide whether the model (*i.e.*, assistant role) should interrupt the video and start its own turn. Compared with VideoLLM-Online[32] that makes this decision by predicting a special token using the LM Head, our design of using two additional heads has the following merits: (1) By combining two scores we can flexibly set different criteria for response generation, rather than only relying on the logits of the special tokens; (2) The relevance head can be used to precisely perform temporal video grounding and highlight detection tasks, expanding the application scenarios of MMDuet.

4.2. Inference Procedure

A python-style pseudocode of the inference process is shown in Listing 1. When consuming every single sampled frame of the video, we first check if there is a user query happening at this time. If yes, we first input this user turn to the model. Then the sampled frame is input to the model, after which the informative score and relevance score are calculated. In addition, we use a function `need_response` to estimate whether the model should generate an assistant response according to the informative scores and relevance scores for this frame along with previous frames. If yes, the `generate` function of the LLM outputs a response. Different `need_response` functions can be designed depending on the specific task, which is introduced in the experiment section (Sec. 6). This process can be efficiently implemented by updating the KV Cache each time when a frame or text is input or generated.

```
# Input:
#  system_prompt
#  video: list of frames
#  fps: frames per second to sample from video
#  user_turns: list of (time, text) sorted by time
# Output:
#  model_turns: generated list of (time, text)

model_turns = []
v_inf_list, v_rel_list = [], []
kv_cache = model(system_prompt)
time = 0
for frame in video:
  if len(user_turns) and time>=user_turns[0].time:
    kv_cache = model(kv_cache, user_turns[0].text)
    user_turns = user_turns[1:]
  kv_cache, v_inf, v_rel = model(kv_cache, frame)
  v_inf_list.append(v_inf)  # informative score
  v_rel_list.append(v_rel)  # relevance score
  if need_response(v_inf_list, v_rel_list):
    kv_cache, response = model.generate(kv_cache)
    model_turns.append((time, response))
  time += 1 / fps
```

Listing 1. Inference Process of MMDuet

5. MMDuetIT: Dataset for Training MMDuet

We build MMDuetIT, a dataset for training the MMDuet model to learn to calculate the informative and relevance scores, and autonomously output replies at any necessary time in the play of the video. MMDuetIT is composed of three different types of tasks that benefit our model training: dense captioning, multi-answer grounded video question answering, and temporal video grounding. An example of the input format for each task is listed in the appendix.
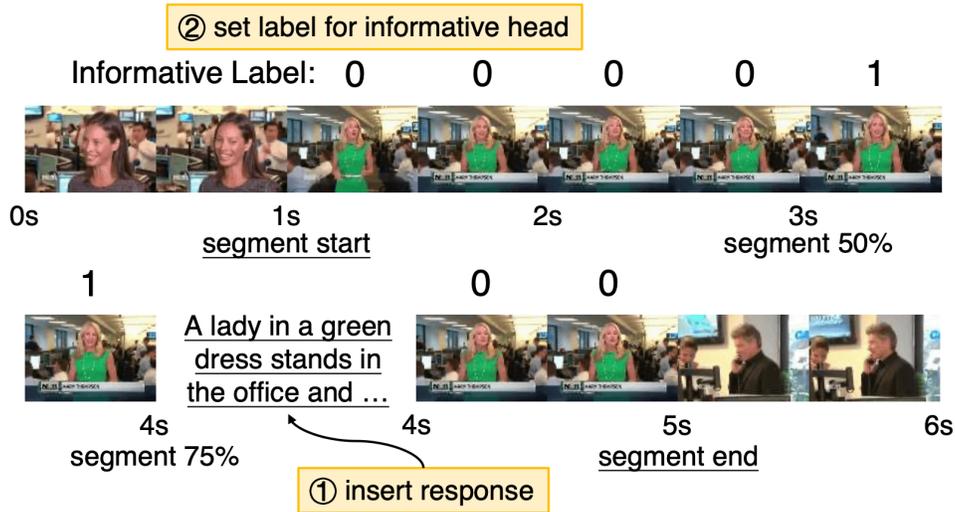
Figure 2. Example of reformatting the annotation of a video segment to video-text duet interaction format in MMDuetIT. Information from the original annotation is emphasized with underlines.

We use Shot2Story[35], a video-text dataset with segment-level captions, as our dense captioning training data. Specifically, we use the 43k human-annotated subset due to its high-quality and detailed annotations. We preprocess the data to serve our purposes, and an illustration of reformatting the video segment and caption annotations to video-text duet interaction format is in Fig. 2.

Choices of insertion

We randomly sample a position from 50% to 75% time duration for the corresponding video segment, and insert the caption at that position as a model response. Here we introduce some randomness in the insertion position to prevent the model from developing a bias or a shortcut such as responses can only be generated at some specific positions. The earliest and latest time for inserting responses, *i.e.*, at the 50% and 75% place of segment duration, are empirically chosen, as it works well in our preliminary study. We avoid inserting responses too early like in the first half of duration, because it is unfeasible to generate responses related to this video segment at a very starting point. It is reasonable that some further observations are required to gain a more comprehensive understanding of it. We also avoid inserting responses too late like in the last one-fourth duration, as we hope the model to

output a response as soon as it has a sufficient understanding of the segment, rather than wait until the disappearance of the segment. It thereby improves the timeliness of the whole interaction between users and videos, especially when the user can still watch the segment as well as perceive the content of the model response talking about it.

Creating informative labels

We also create labels for the informative head in dense captioning tasks. According to the previous paragraph, the model can not have a comprehensive understanding of this video segment until it has viewed a sufficient portion of the segment (50% in this case). Meanwhile, once the caption has been generated as model response, we assume that the remaining frames in this video segment no longer provide new information that is not covered in the caption. Therefore, we set the informative head's label to *TRUE* for frames between 50% of this segment and the insertion point of the response, and set labels to *FALSE* for the other frames, as shown in Fig. 2.

To adapt to long video input, we also select videos with 2 to 4 minutes in length from COIN[34] as a dense captioning task to MMDuetIT. The annotations in COIN are reformatted using the same method as Shot2Story.

5.2. Multi-Answer Grounded Video QA

An important application scenario for the video-text duet interaction format is multi-answer grounded video question-answering (MAGQA). Consider when we are watching a live broadcast of a basketball game and want to track the actions of a particular player in the game. This exemplifies a MAGQA task: the question is "What does this particular player do in the video?". When this player performs an action, the model should respond with a description of this action (i.e., multiple answers) in a real time manner. We believe this newly proposed MAGQA task can be widely used in real-world scenarios when users interact with a live-streaming video.

We construct training data for this task using GPT4o-2024-08-06[36]. Given the captions of all segments from the video as input, GPT4o is prompted to generate a question related to one or more captions. For each of the segment captions, if it is related to the question, then GPT4o should also generate an answer that can be inferred from this caption. Otherwise, GPT4o should reply with "Not Mentioned.", and this answer is not added to the training data. We use the same insertion method of dense captioning task as described in 5.1, to insert the answers into the video stream and construct

informative head labels, and the question is inserted at a random place before the first answer. We also use the same insertion method to convert the human-annotated Shot2Story test set and randomly sampled 2000 examples as the test set of our MAGQA benchmark in 6.3. Therefore, this dataset contains 36834 examples in the train set and 2000 examples in the test set. We name it as "Shot2Story-MAGQA-39k" and show its statistics in Table 1.

Data Quality Assessment

We sample 100 examples (with 290 answers) from our test set for manual quality assessment. Among the sampled examples, we find 1 example with a question unanswerable from the video, 5 examples have 6 answers (2.1%) that contradict the video content, and 5 examples have 7 answers (2.4%) unrelated to the question. Overall, manual quality assessment shows that above 95% data of our test set belongs to the high quality, which confirms the potential value of using Shot2Story-MAGQA-39k to benchmark models. The reason for the high quality is when the video captions are provided, generating questions and answers based on these text captions is a very simple task for advanced LLMs like GPT4o. However, we also find that in 21 examples, the video contains additional information that is not covered in the answers. This is because some questions are very general, like "What scene is the video displaying?", and describing scenes in videos elaborately has been a long-lasting challenge for annotating video datasets.

5.3. Temporal Video Grounding

We also add DiDeMo[19], HiREST$_{grounding}$[37] and QuerYD[38], three temporal video grounding tasks in MMDuetIT. Note that these data are used only for training the relevance head, which is designed for performing temporal video grounding tasks and judging the relevance between the question and the video for QA tasks. The query is first added at the beginning of the input sequence. For frames that are annotated as relevant to the query, we set the relevance head's label to *TRUE*; otherwise, we set it to *FALSE*.

5.4. Dataset Statistics

The data distribution of MMDuetIT is shown in Fig. 3. Note that this dataset only contains 109k examples, which is relatively small compared to modern post-training datasets like[11][14][31]. The reason is that due to computational resource constraints, we plan to demonstrate the feasibility of our

proposed video-text duet interaction format by fine-tuning a state-of-the-art VideoLLM. We assume that the used backbone model already possesses enough video comprehension capabilities. By using a small dataset, we aim to train this model to efficiently adopt this new interaction with minimum catastrophic forgetting of its existing abilities.

	num examples	answers per video	words per ques./ans.	video seg. len (sec)
Train	36834	2.96	7.75/12.17	4.22
Test	2000	3.04	7.77/12.17	4.28

Table 1. Dataset statistics of Shot2story-MAGQA-39k.
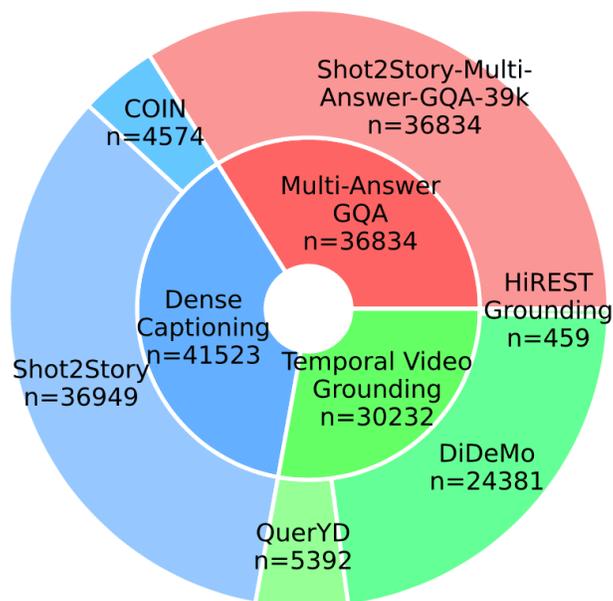
Figure 3. Data Distribution of MMDuetIT.

	QVHighlights	Charades-STA
	mAP/HIT@1	R@0.5/0.7
TimeChat	14.5/23.9	32.2/13.4
VTimeLLM	-	31.2/11.4
HawkEye	-	31.4/14.5
VTG-LLM	16.5/33.5	33.8/15.7
LLaVA-OV-TC	17.6/32.9	33.1/12.4
LLaVA-OV-VT	19.0/40.0	36.5/12.3
MMDuet (Ours)	**31.3/49.6**	**42.4/18.0**

Table 2. Zero-shot performance on highlight detection and temporal video grounding.

	YouCook2
	SODA/c/CIDEr/F1
TimeChat	1.2/3.4/12.6
VTG-LLM	1.5/5.0/17.5
LLaVA-OV-TC	1.9/3.3/**21.8**
LLaVA-OV-VT	2.5/6.7/14.0
MMDuet (Ours)	2.4/5.7/19.2
+ rm. prev. resp.	**2.9/8.8/21.7**

Table 3. Zero-shot performance on dense video captioning.

6. Experiments

Implementations

MMDuet is initialized with LLaVA-OneVision[14]. We train the model on MMDuetIT for one epoch. The training takes about one day on a node with 8 Tesla V100 GPUs, and the inference runs on 1 Tesla V100 GPU. Implementation details are listed in the appendix.

Baselines

As MMDuet mainly focuses on time-sensitive video tasks, we use the following baselines that are able to represent time spans in videos by different representation formats: TimeChat[27], VTimeLLM (7B) [28], HawkEye[31] and VTG-LLM[29]. Since the initialization of MMDuet is stronger than that of the baselines, for a fair comparison we also conduct a controlled experiment in which the only difference is the interaction format. Specifically, we use the same initialization model (LLaVA-OneVision), training data (MMDuetIT) and training schedule, but reformat the data to the respective interaction formats and video segment representation formats used by TimeChat and VTimeLLM to train two baseline models. We refer to these models as LLaVA-OV-TC and LLaVA-OV-VT.

6.1. Highlight Detection and Temporal Video Grounding

We use highlight detection and temporal video grounding to evaluate the performance of the relevance head of MMDuet. Baseline models are required to generate a list of float numbers to represent the relevance score for each clip in QVHighlights[20], and a start and end time for the relevant video span in Charades-STA. However, for LLaVA-OV-TC and LLaVA-OV-VT, despite using different prompts as input, we were still unable to instruct the model to output a sequence of scores as in[27]. Therefore, we follow the method of Charades-STA to instruct the model to output a related span, and assign the score to 1 for clips within this span and 0 otherwise. MMDuet uses the relevance score min-max normalized to $[0, 1]$ as the score in QVHighlights, and to classify whether this frame is relevant and calculate frame-level IoU in Charades-STA.

Since the relevance head provides a relevance score immediately after each frame, its prediction cannot leverage the context from subsequent video frames. To mitigate this limitation, we smooth the relevance score sequence. Specifically, we set each frame's smoothed relevance score as the mean

value of its original score, the relevance scores of the preceding w frames and the following w frames, where w is the window size. We set $w = 2$ for QVHighlights and $w = 6$ for Charades-STA. Results are shown in Tab. 2. We observe that, compared to the baselines, MMDuet exhibits a significantly greater improvement in performance on QVHighlights. This indicates that traditional VideoLLMs struggle with generating a long sequence of relevance scores using a text-based form or identifying multiple related video segments in its text-based responses, whereas MMDuet's approach of directly assigning relevance scores to each frame circumvents this issue.

Parameter sensitivity of w

Though the w is empirically set for the results in Tab. 2, in Fig. 4 we show that within a fairly large range of w, MMDuet consistently outperforms all baseline models. It confirms that MMDuet is robust to these tasks with respect to the window size.
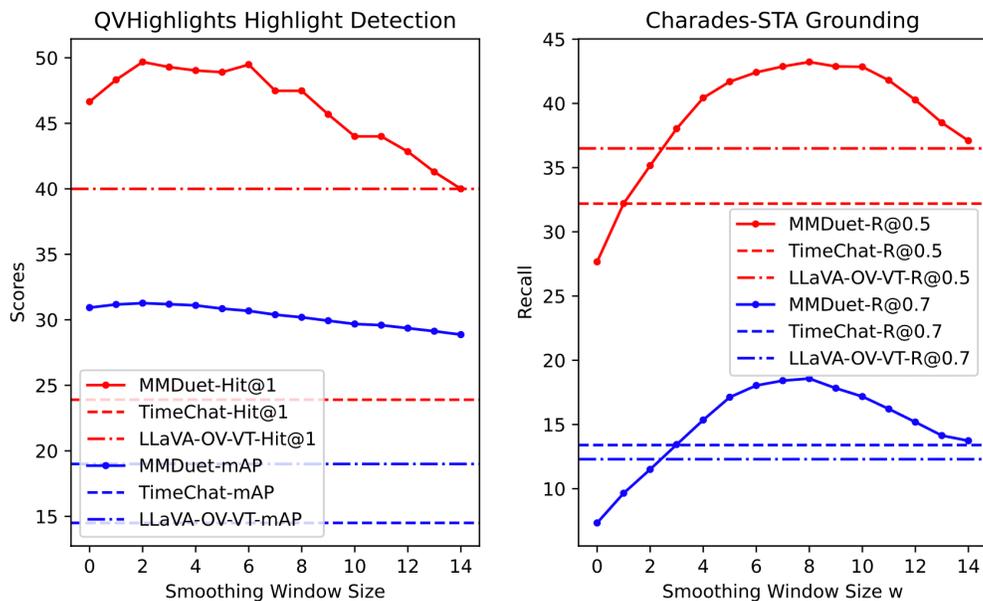
Figure 4. Performance on temporal video grounding and highlight detection with different w.

6.2. Dense Video Captioning

We test dense video captioning performance on YouCook2[21], a challenging task that requires models to output the caption, start point and end point for about 8 steps in a minutes-long cooking video.

Baseline models output the start time, end time and caption for each step in the text-based form. For MMDuet, since this task requires the model to continuously identify important actions from the video and output periodically, we employ a heuristic method to determine whether a model response should be output after each frame (`need_response` function in Sec. 4.2). We sum up the informative score for each frame as the video plays. When the sum reaches a threshold s (we set $s = 2$), the model generates a response right after this frame as the caption for that step, and then we reset the sum to 0 to start a new round of sum.

However, MMDuet cannot directly predict when a step starts or ends just by this video-text duet interaction format, as the model is unable to determine whether a frame is the beginning of a step without observing enough subsequent content. To get the start and end time for each step as required by this task, we adopt a simple workaround: we use the time of the previous response and the current response as the start time and end time for a step. If two adjacent steps have the same caption, we merge them into one step.

It has been a long-lasting problem that LLMs tend to repeat previously-generated content[39], and we find that this problem is especially severe in dense video captioning. It indicates that VideoLLMs are probably generating captions relying on text shortcuts rather than the video content. We have attempted common solutions such as repetition penalty[40], which though is still sub-optimal. Since the responses from MMDuet are separated across multiple turns, we find that simply removing previously generated turns from the context ("rm. prev. resp." for short) by not appending their attention keys and values to the KV Cache alleviates this issue, leading to a significant improvement in performance.

As shown in Tab. 3, MMDuet does not show significant improvements on F1 metric, likely due to the simple solution we use to derive the start and end time based on responses. Even so, the CIDEr and CODA_c metric (inaccurate predicted time spans can have negative effects on these metrics) of MMDuet is still higher than all baselines, indicating that MMDuet outperforms baselines in terms of text quality, possibly due to its facilitation to information retrieval discussed in Sec. 1.

Parameter sensitivity of s

We report the performance with threshold s from 1 to 3 in Fig. 5. Results show that the performance is quite robust across a wide range of s, and we simply choose $s = 2$ in Tab. 3. Note that we can use different s to suit various downstream tasks especially in such zero-shot setting.
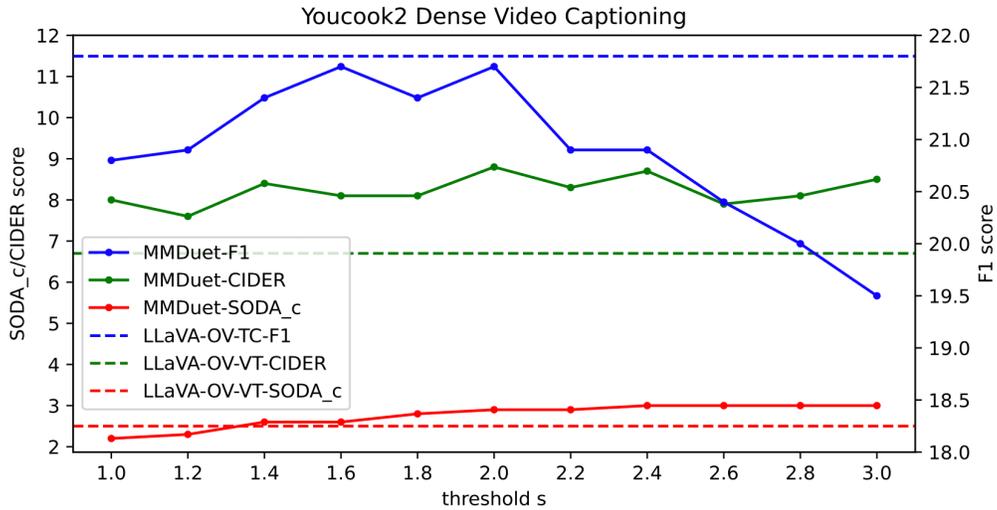
Figure 5. Performance on dense video captioning with different s.

Model	Real-Time?	In-Span Score	# turns (w/o. / w/. dedup)	time per example
Baselines				
LLaVA–OV–TC	✗	2.92/2.79	3.4/1.9	0.76
LLaVA–OV–VT	✗	2.94/2.78	5.4/2.2	1.00
MMDuet (Ours)				
$t = 0.6$	✔	2.46/2.33	13.7/4.0	1.80
$t = 0.5$	✔	2.77/2.61	18.4/5.3	2.23
$t = 0.4$	✔	3.00/2.81	23.0/6.6	2.59
$t = 0.3$	✔	3.13/2.93	27.0/7.6	2.73

Table 4. Results on the test set of Shot2Story–MAGQA–39k with the rm. ass. turns method used. For the "time per example" column, the time used by "LLaVA–OV–VT" is set to 1, and the times for other rows are set as multiples of the time used by "LLaVA–OV–VT".

6.3. Multi-Answer Grounded Video QA

To align closely with the widely-used streaming video comprehension scenario, we propose MAGQA that requires a model to generate answers at multiple necessary positions of a video. Different from conventional Video QA in which one question corresponds to only one answer, In MAGQA, a question corresponds to multiple turns of answers, and these turns are derived from different video segments. Therefore, this task requires the response to be accurate and in time.

We introduce an "in-span score" metric, which takes both the text content and reply time of the predicted answer into account to evaluate the model's performance on this task. Suppose the model prediction has P answers, each answer has a prediction time $time_p$ and prediction text $pred_p$, $p = 1, 2, \ldots, P$. The ground truth has Q answers, each answer has a ground truth start time $start_q$, a ground truth end time end_q, and a ground truth text $gold_q$, $q = 1, 2, \ldots, Q$. First, we use an LLM to calculate a relevance score from 1 to 5 between each answer in prediction $pred_p$ and ground truth $gold_q$: $S = \{s_{p,q}\} \in \mathbb{R}^{P \times Q}$. For each ground truth answer q, we select the predicted answers with predicted time in ground truth time span: $\mathcal{P}_q = \{p \mid time_p \in [start_q, end_q]\}$, and use the average score between the ground truth answer and the selected predicted answers as the score for this ground truth answer: $score_q = \frac{1}{|\mathcal{P}_q|} \sum_{p \in \mathcal{P}_q} s_{p,q}$ if $|\mathcal{P}_q| > 0$. If $|\mathcal{P}_q| = 0$ (no predicted answer falls in this ground truth span), $score_q$ is set to 1. Finally, we calculate the average score of all ground truth answers as the final in-span score of this example: $in\_span\_score = \frac{1}{|Q|} \sum_{q=1}^{|Q|} score_q$. GPT-4o-2024-08-06[36] is used as the scorer LLM for in-span score. To prevent reproducibility issues due to potential changes of OpenAI API, we also report the in-span score obtained using LLaMA 3.1 70B Instruct[4].

As MAGQA requires the answers to be both informative and related to the question, we set the *need_response* as: if the sum of informative score and relevance score of a frame is larger than a threshold t, then the model needs to generate a response right after this frame. We also use the "rm. prev. resp." method in dense video captioning task introduced in Sec. 6.2. As baseline models are not capable of generating responses at specific positions in the video, we employ an output format the same as dense video captioning, i.e., output the start time, end time, and predicted text for each turn after watching the entire video in both training and testing, and use the average of the start and end time as the response time. For the cases that baseline models directly reply with an answer without corresponding time, we pair this predicted answer with all ground truth answers. **Note that this is a simplified requirement than that of MMDuet,** as the MAGQA task simulates streaming video comprehension application scenario, which requires the model to respond as soon as the video plays

to segments relevant to the question, which ensures that users can see the responses timely, rather than waiting until the entire video concludes before generating replies.

Results are shown Tab. 4. We provide results for different t as it represents a trade-off between inference time and performance: as t decreases from 0.6 to 0.3, the performance of MMDuet's real-time replies continuously rise and even outperform baselines with a simplified setting of providing non-real-time replies after watching the entire video. However, this is achieved at a cost of generating lots of duplicate replies with around $2.5\times$ inference time.

Model	YouCook2
MMDuet	2.9/8.8/21.7
w/o rand. resp. pos.	2.1/7.3/19.0
w/o multi informative	2.9/8.0/16.5

Table 5. Ablation study on training methods.

6.4. Ablation Studies

We conduct ablation studies on YouCook2 dense video captioning to assess two empirical yet important findings for effectively training the informative head in data construction: randomly inserting the response at a position from 50% to 75% of the corresponding video segment (rand. resp. pos.), and setting informative head's label to TRUE for all frames between 50% of the segment and the response time (multi informative). When "rand. resp. pos." is disabled, the response is always inserted at the end of the corresponding segment. When "multi informative" is disabled, only the informative label of the frame right before the response is set as TRUE. As illustrated in Tab. 5, disabling either method negatively impact MMDuet's performance, which shows the importance of carefully handling the response time and informative labels.

7. Limitations and Future Works

As a preliminary and pioneering effort of video-text duet interaction format, we acknowledge that there is much room for improvement which should be addressed in future research: (1) Some

hyperparameters (e.g., the *need_response* criterion) are required during inference. Though the parameter sensitivity test shows that this criterion is quite robust across different thresholds, there still exists unknowns for applying MMDuet on new tasks. (2) Information from subsequent frames is not incorporated when generating in-time responses for the current frame, especially for the live-streaming video that indeed has unpredictable future frames. It can be crucial in some scenarios, such as determining the start of an action. (3) Slow inference speed. A better inference process is needed to improve the parallelism of video inputs and avoid generating duplicate responses. (4) Real-time response datasets with longer live-streaming videos are required to be collected to better fit the real-world application scenarios.

8. Conclusion

In this paper, we formalize the video-text duet interaction format, which addresses the shortcomings of existing interaction formats in terms of time-sensitive task performance, more application scenarios and in-time information retrieval for long videos. We collect MMDuetIT, a dataset for training models to follow the video-text duet interaction format. Based on MMDuetIT we train MMDuet, a model with significant improvements on various time-sensitive tasks, and is able to generate responses in real time. We believe such improvements can be a substantial step towards building powerful and useful video comprehension systems.

Appendix A. Details of Training MMDuet

A.1. Training Hyperparameters

LLaVA-OneVision uses SigLIP-Large[7] as the vision encoder, and converts an image with 384×384 into $24 \times 24 = 576$ tokens. In the official settings of LLaVA-OneVision[14], when encoding videos, the visual tokens corresponding to each frame are spatially downsampled to $12 \times 12 = 144$ tokens using a pooling operation with a size of 2. However, this number of tokens is also too large when training and inference with long videos. To address this, we further modified the pooling size to 4, resulting in $7 \times 7 = 49$ tokens per frame.

We set the maximum number of frames sampled from each video to 120 in the training process, which is constrained by the memory of our GPUs. The sampling frame rates are set to different numbers for different video sources to ensure that for the vast majority (>90%) of videos, video length (in

seconds) ÷ sampled frame per second (fps) ≤ 120. For the videos that are too long, we only keep the first 120 frames (and the conversation turns that are inserted within the first 120 frames), and discard the subsequent contents. Specifically, the sampled frame per second (fps) is set as: 2 for videos from Shot2Story[35] and DiDeMo[19], 0.5 for COIN[34] and QueryD[38], and 0.33 for HiREST$_{grounding}$ [37].

The projector, the relevance head, the informative head and LoRA[41] weights of the LLM (add to all attention proj. layers and FFN layers) are trained, while other parameters of the model are frozen. More training hyperparameters are listed in 6.

A.2. Inference Settings

Videos from different sources are also sampled with different fps during inference. Specifically, we set the maximum number of frames sampled from each video to 400, and fps to 2 for videos from Shot2Story[35] and Charades-STA[18], 1 for videos from QVHighlights[20], and 0.5 for videos from YouCook2[21]. For a few videos in YouCook2 that are even longer than $400(\text{frames}) \div 0.5(\text{fps}) = 800$ seconds, we uniformly sample 400 frames from this video to ensure that information from the latter part of the video is not truncated. This inference setting is consistent across MMDuet, LLaVA-OV-TC, and LLaVA-OV-VT.

Hyper-parameter	value
batch_size	1
gradient_acc_steps	8
learning_rate	2e-5
warmup_ratio	0.05
lora_r	16
lora_alpha	32
attn_implementation	sdpa

Table 6. Hyper-parameters used for training MMDuet.

Appendix B. Example Inputs for Each Task in MMDuetIT

Example inputs for each task for training and inference are listed in Tab. 7. The dense video captioning user input is selected from one of the following sentences:

- Please concisely narrate the video in real time.
- Help me to illustrate my view in short.
- Please simply describe what do you see.
- Continuously answer what you observed with simple text.
- Do concise real-time narration.
- Hey assistant, do you know the current video content? Reply me concisely.
- Simply interpret the scene for me.
- What can you tell me about? Be concise.
- Use simple text to explain what is shown in front of me.
- What is the action now? Please response in short.

The temporal video grounding user input is selected from one of the following sentences (where "%s" denotes the caption to localize):

- %s What segment of the video addresses the topic '%s'?
- At what timestamp can I find information about '%s' in the video?
- Can you highlight the section of the video that pertains to '%s'?
- Which moments in the video discuss '%s' in detail?
- Identify the parts that mention '%s'.
- Where in the video is '%s' demonstrated or explained?
- What parts are relevant to the concept of '%s'?
- Which clips in the video relate to the query '%s'?
- Can you point out the video segments that cover '%s'?
- What are the key timestamps in the video for the topic '%s'?

Dense Video Captioning	`<im_start>`system A multimodal AI assistant is helping users with some activities. Below is their conversation, interleaved with the list of video frames received by the assistant. `<im_end>` `<im_start>`user (A Dense Video Captioning Query)`<im_end>` `<im_start>`stream `<frame><frame><frame>` ...`<im_end>` `<im_start>`assistant A person pulls a knife from a black bag.`<im_end>` `<im_start>`stream `<frame><frame><frame>` ...`<im_end>` `<im_start>`assistant A man in a hat and red clothes speaks with a dagger, and a tree behind him.`<im_end>` `<im_start>`stream `<frame><frame><frame>` ...`<im_end>` ...(More stream and assistant turns)
MAGQA	`<im_start>`system A multimodal AI assistant is helping users with some activities. Below is their conversation, interleaved with the list of video frames received by the assistant. `<im_end>` `<im_start>`stream `<frame><frame><frame>` ...`<im_end>` `<im_start>`user What happens during the basketball game?`<im_end>` `<im_start>`stream `<frame><frame><frame>` ...`<im_end>` `<im_start>`assistant Several players in white jerseys are celebrating by high-fiving each other.`<im_end>` `<im_start>`stream `<frame><frame><frame>` ...`<im_end>` `<im_start>`assistant A player in a white jersey makes a successful shot.`<im_end>` `<im_start>`stream `<frame><frame><frame>` ...`<im_end>` ...(More stream and assistant turns)
Temporal Video Grounding	`<im_start>`system A multimodal AI assistant is helping users with some activities. Below is their conversation, interleaved with the list of video frames received by the assistant. `<im_end>` `<im_start>`user (A Temporal Video Grounding Query)`<im_end>` `<im_start>`stream `<frame><frame><frame>` ...`<im_end>`

Table 7. Input examples of different tasks during the training and evaluation phase of MMDuet.

Appendix C. Qualitative Study

We list some examples of dense video captioning on videos with several minutes in length and contains many actions in Figs. 6-8, and examples of multi-answer grounding video question answering (MAGQA) in Figs. 9-11. For LLaVA-OV-TC and LLaVA-OV-VT, we directly list their generated outputs. For MMDuet, we list the numerical order (in round brackets), time (in square brackets) and content (in the second line) for each turn. If a line contains multiple numerical orders and times, this indicates that these turns have the same content, which is shown in the following line.

To help readers to identify the position of these turns within the video, we also annotate the numerical order of the turns at the corresponding timestamps in the video stream.

When handling long videos for dense video captioning, baseline models often recall only part of the video or generate repeated content, failing to provide a complete description of all steps in the video. In contrast, MMDuet, due to its ability to focus only on a small portion of the video content preceding each generation step and using the "rm. prev. turns" trick to avoid interference from previous turns, can provide more accurate and detailed video descriptions.

For the MAGQA task, due to the relatively short video length, baseline models can also locate video segments and answer questions effectively. The advantage of MMDuet in this task is its ability to provide answers in a real-time manner.

Figure 6. An example of dense video captioning with MMDuet, LLaVA-OV-TC and LLaVA-OV-VT.

Figure 7. An example of dense video captioning with MMDuet, LLaVA-OV-TC and LLaVA-OV-VT.

MMDuet (Ours)
(0) [0s]
USER: What can you tell me about? Be concise.
(1) [44.0s] (2) [66.0s]
ASSISTANT: pour the noodles into the water
(3) [96.0s]
ASSISTANT: prepare the ingredients
(4) [126.0s]
ASSISTANT: pour the sauce into the pan
(5) [150.0s]
ASSISTANT: pour the oil into the pan
(6) [170.0s]
ASSISTANT: pour the raw materials into the pot
(7) [184.0s]
ASSISTANT: put the noodles into the pot
(8) [200.0s]
ASSISTANT: pour the noodles into the pot
(9) [214.0s]
ASSISTANT: add seasoning
(10) [228.0s]
ASSISTANT: pour sauce
(11) [240.0s]
ASSISTANT: pour the noodles into the pot
(12) [258.0s]
ASSISTANT: add seasoning
(13) [278.0s]
ASSISTANT: put the food on the plate
(14) [296.0s] (15) [308.0s]
ASSISTANT: add seasoning

LLaVA-OV-TC
10.0 seconds, Intro.
25.0 seconds, Showing noodles.
34.0 seconds, Boiling noodles.
67.0 seconds, Showing ingredients.
89.0 seconds, Showing sauce.
108.0 seconds, Cooking noodles.
130.0 seconds, Adding shrimp.
150.0 seconds, Adding tofu.
165.0 seconds, Adding eggs.
180.0 seconds, Stirring noodles.
200.0 seconds, Seasoning noodles.
220.0 seconds, Plating dish.
240.0 seconds, Outro.

LLaVA-OV-VT
From 00 to 31, pour the noodles into the water.
From 31 to 42, fry the ingredients in the pan.

Figure 8. An example of dense video captioning with MMDuet, LLaVA-OV-TC and LLaVA-OV-VT.
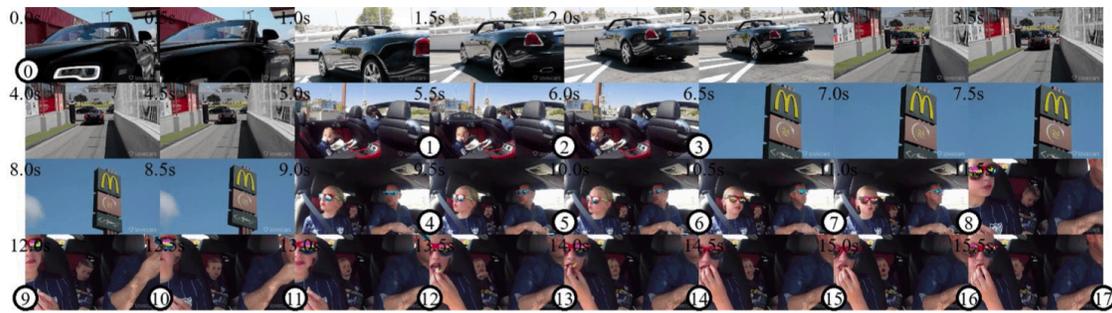
Question

What are the people doing in the car?

MMDuet (Ours)

(0) [0s]
USER: What are the people doing in the car?
(1) [5.5s] (2) [6.0s] (3) [6.5s]
ASSISTANT: A man is driving the car and a boy is sitting in the back seat.
(4) [9.5s]
ASSISTANT: A man and a child are sitting in the car, with the man driving and the child in the back seat.
(5) [10.0s]
ASSISTANT: A man and a child are sitting in the car, with the man driving and the child in the passenger seat.
(6) [10.5s]
ASSISTANT: A man and a child are sitting in the car, wearing sunglasses.
(7) [11.0s] (8) [11.5s]
ASSISTANT: A man and a child are sitting in the car, and the man is driving.
(9) [12.0s]
ASSISTANT: A man and a boy are sitting in the car, and the boy is wearing sunglasses.
(10) [12.5s] (11) [13.0s] (12) [13.5s] (13) [14.0s] (14) [14.5s] (15) [15.0s] (16) [15.5s]
ASSISTANT: A man and a boy are sitting in the car, and the boy is eating something.
(17) [16.0s]
ASSISTANT: A man and a child are sitting in the car, and the man is driving.

LLaVA-OV-TC

0.0 - 5.0 seconds, A man is driving a black convertible car with a child in the back seat.
5.0 - 6.5 seconds, A man is driving a black convertible car with a child in the back seat.
6.5 - 10.0 seconds, A man and a child are sitting in the back seat of a car, with the child eating something.
10.0 - 16.5 seconds, A man and a child are sitting in the back seat of a car, with the child eating something.

LLaVA-OV-VT

From 00 to 10, A man is driving a black convertible car with the top down.
From 10 to 20, A man is driving a black convertible car with the top down.
From 20 to 31, A man is driving a black convertible car with the top down.
From 31 to 41, A man is driving a black convertible car with the top down, and a child is sitting in the back seat.
From 41 to 51, A man is driving a black convertible car with the top down, and a child is sitting in the back seat.
From 51 to 94, A man is driving a black convertible car with the top down, and a child is sitting in the back seat.

Figure 9. An example of multi-answer grounded video question answering with MMDuet, LLaVA-OV-TC and LLaVA-OV-VT.
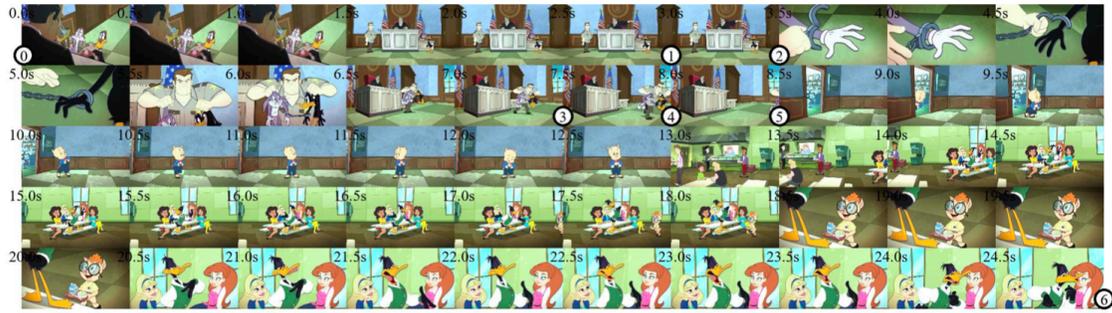
Question
What happens to the rabbit and the duck?

MMDuet (Ours)
(0) [0s]
USER: What happens to the rabbit and the duck?
(1) [3.0s] (2) [3.5s]
ASSISTANT: The rabbit and the duck are sitting at a table in a courtroom.
(3) [7.5s]
ASSISTANT: The rabbit and the duck are held by a policeman.
(4) [8.0s] (5) [8.5s]
ASSISTANT: The rabbit and the duck are held by a man in a police uniform and run away.
(6) [25.0s]
ASSISTANT: The duck and the rabbit are sitting at a table in a restaurant.

LLaVA-OV-TC
8.7 - 11.0 seconds, The rabbit and the duck are tied up by a man in a police uniform.

LLaVA-OV-VT
From 00 to 11, The rabbit and the duck are sitting at a table in a courtroom, with the rabbit wearing a purple shirt and the duck wearing a black shirt.
From 11 to 20, The rabbit and the duck are sitting at a table in a courtroom, with the rabbit wearing a purple shirt and the duck wearing a black shirt.
From 20 to 25, The rabbit and the duck are sitting at a table in a courtroom, with the rabbit wearing a purple shirt and the duck wearing a black shirt.
From 25 to 31, The rabbit and the duck are sitting at a table in a courtroom, with the rabbit wearing a purple shirt and the duck wearing a black shirt.
From 31 to 35, The rabbit and the duck are sitting at a table in a courtroom, with the rabbit wearing a purple shirt and the duck wearing a black shirt.
From 35 to 41, The rabbit and the duck are sitting at a table in a courtroom, with the rabbit wearing a purple shirt and the duck wearing a black shirt.
From 41 to 50, The rabbit and the duck are sitting at a table in a courtroom, with the rabbit wearing a purple shirt and the duck wearing a black shirt.
From 50 to 60, The rabbit and the duck are sitting at a table in a cafeteria, with the rabbit wearing a purple shirt and the duck wearing a black shirt.
From 60 to 65, The rabbit and the duck are sitting at a table in a cafeteria, with the rabbit wearing a purple shirt and the duck wearing a black shirt.
From 65 to 70, The rabbit and the duck are sitting at a table in a cafeteria, with the rabbit wearing a purple shirt and the duck wearing a black shirt.
From 70 to 75, The rabbit and the duck are sitting at a table in a cafeteria, with the rabbit wearing a purple shirt and the duck wearing a black shirt.
From 75 to 81, The rabbit and the duck are sitting at a table in a cafeteria, with the rabbit wearing a purple shirt and the duck wearing a black shirt.
From 81 to 96, The rabbit and the duck are sitting at a table in a cafeteria, with the rabbit wearing a purple shirt and the duck wearing a black shirt.

Figure 10. An example of multi-answer grounded video question answering with MMDuet, LLaVA-OV-TC and LLaVA-OV-VT.

Figure 11. An example of multi-answer grounded video question answering with MMDuet, LLaVA-OV-TC and LLaVA-OV-VT.

References

1. △Touvron H, Martin L, Stone KR, Albert P, Almahairi A, Babaei Y, Bashlykov N, Batra S, Bhargava P, Bho sale S, Bikel DM, Blecher L, Cant{\'o}n Ferrer C, Chen M, Cucurull G, Esiobu D, Fernandes J, Fu J, Fu W, F uller B, Gao C, Goswami V, Goyal N, Hartshorn AS, Hosseini S, Hou R, Inan H, Kardas M, Kerkez V, Khabs a M, Kloumann IM, Korenev AV, Koura PS, Lachaux MA, Lavril T, Lee J, Liskovich D, Lu Y, Mao Y, Martin et X, Mihaylov T, Mishra P, Molybog I, Nie Y, Poulton A, Reizenstein J, Rungta R, Saladi K, Schelten A, Sil va R, Smith EM, Subramanian R, Tan X, Tang B, Taylor R, Williams A, Kuan JX, Xu P, Yan Z, Zarov I, Zha ng Y, Fan A, Kambadur M, Narang S, Rodriguez A, Stojnic R, Edunov S, Scialom T. Llama 2: Open found ation and fine-tuned chat models. ArXiv. 2023; **abs/2307.09288**.

2. △Jiang AQ, Sablayrolles A, Mensch A, Bamford C, Chaplot DS, de Las Casas D, Bressand F, Lengyel G, La mple G, Saulnier L, Renard Lavaud L, Lachaux MA, Stock P, Le Scao T, Lavril T, Wang T, Lacroix T, El Sa yed W (2023). "Mistral 7B". ArXiv. **abs/2310.06825**.

3. ^*Shao Z, Dai D, Guo D, Liu B (Benjamin Liu), Wang Z, Xin H (2024). "DeepSeek-V2: A Strong, Economical, and Efficient Mixture-of-Experts Language Model". ArXiv. **abs/2405.04434**. Available from: https://arxiv.org/abs/2405.04434.*

4. ^a, ^b *Dubey A, Jauhri A, Pandey A, Kadian A, Al-Dahle A, Letman A, Mathur A, Schelten A, Yang A, Fan A, Goyal A, Hartshorn A, Yang A, Mitra A, Sravankumar A, Korenev A, Hinsvark A, Rao A, Zhang A, Rodriguez A, Gregerson A, Spataru A, Rozière B, Biron B, Tang B, Chern B, Caucheteux C, Nayak C, Bi C, Marra C, McConnell C, Keller C, Touret C, Wu C, Wong C, Cantón Ferrer C, Nikolaidis C, Allonsius D, Song D, Pintz D, Livshits D, Esiobu D, Choudhary D, Mahajan D, Garcia-Olano D, Perino D, Hupkes D, Lakomkin E, AlBadawy EA, Lobanova E, Dinan E, Smith EM, Radenovic F, Zhang F, Synnaeve G, Lee G, Anderson GL, Nail G, Mialon G, Pang G, Cucurell G, Nguyen H, Korevaar H, Xu H, Touvron H, Zarov I, Arrieta Ibarra I, Kloumann IM, Misra I, Evtimov I, Copet J, Lee J, Geffert JL, Vranes J, Park J, Mahadeokar J, Shah J, van der Linde J, Billock J, Hong J, Lee J, Fu J, Chi J, Huang J, Liu J, Wang J, Yu J, Bitton J, Spisak J, Park J, Rocca J, Johnstun J, Saxe J, Jia J, Alwala KV, Upasani K, Plawiak K, Li K, Heafield K, Stone K, El-Arini K, Iyer K, Malik K, Chiu K, Bhalla K, Rantala-Yeary L, van der Maaten L, Chen L, Tan L, Jenkins L, Martin L, Madaan L, Malo L, Blecher L, Landzaat L, de Oliveira L, Muzzi MC, Pasupuleti MB, Singh M, Paluri M, Kardas M, Oldham M, Rita M, Pavlova M, Hall MK, Lewis M, Si M, Singh MK, Hassan M, Goyal N, Torabi N, Bashlykov N, Bogoychev N, Chatterji NS, Duchenne O, cCelebi O, Alrassy P, Zhang P, Li P, Vasić P, Weng P, Bhargava P, Dubal P, Krishnan P, Koura PS, Xu P, He Q, Dong Q, Srinivasan R, Ganapathy R, Calderer R, Silveira Cabral R, Stojnic R, Raileanu R, Girdhar R, Patel R, Sauvestre R, Polidoro R, Sumbaly R, Taylor R, Silva R, Hou R, Wang R, Hosseini S, Chennabasappa S, Singh S, Bell S, Kim SS, Edunov S, Nie S, Narang S, Raparthy SC, Shen S, Wan S, Bhosale S, Zhang S, Vandenhende S, Batra S, Whitman S, Sootla S, Collot S, Gururangan S, Borodinsky S, Herman T, Fowler T, Sheasha T, Georgiou T, Scialom T, Speckbacher T, Mihaylov T, Xiao T, Karn U, Goswami V, Gupta V, Ramanathan V, Kerkez V, Gonguet V, Do V, Vogeti V, Petrovic V, Chu W, Xiong W, Fu W, Meers W, Martinet X, Wang X, Tan XE, Xie X, Jia X, Wang X, Goldschlag Y, Gaur Y, Babaei Y, Wen Y, Song Y, Zhang Y, Li Y, Mao Y, Delpierre Coudert Z, Yan Z, Chen Z, Papakipos Z, Singh AK, Grattafiori A, Jain A, Kelsey A, Shajnfeld A, Gangidi A, Victoria A, Goldstand A, Menon A, Sharma A, Boesenberg A, Vaughan A, Baevski A, Feinstein A, Kallet A, Sangani A, Yunus A, Lupu A, Alvarado A, Caples A, Gu A, Ho A, Poulton A, Ryan A, Ramchandani A, Franco A, Saraf A, Chowdhury A, Gabriel A, Bharambe A, Eisenman A, Yazdan A, James B, Maurer B, Leonhardi B, Huang B, Loyd B, De Paola B, Paranjape B, Liu B, Wu B, Ni B, Hancock B, Wasti B, Spence B, Stojkovic B, Gamido B, Montalvo B, Parker C, Burton C, Mejia C, Wang C, Kim C, Zhou C, Hu C, Chu C, Cai C, Tindal C, Feichtenhofer C, Civin D, Beaty D, Kre*

ymer D, Li SW, Wyatt D, Adkins D, Xu D, Testuggine D, David D, Parikh D, Liskovich D, Foss D, Wang D, Le D, Holland D, Dowling E, Jamil E, Montgomery E, Presani E, Hahn E, Wood E, Brinkman E, Arcaute E, Dunbar E, Smothers E, Sun F, Kreuk F, Tian F, Ozgenel F, Caggioni F, Guzmán F, Kanayet FJ, Seide F, Medina Florez G, Schwarz G, Badeer G, Swee G, Halpern G, Thattai G, Herman G, Sizov GG, Zhang G, Lakshminarayanan G, Shojanazeri H, Zou H, Wang H, Zha H, Habeeb H, Rudolph H, Suk H, Aspegren H, Goldman H, Molybog I, Tufanov I, Veliche IE, Gat I, Weissman J, Geboski J, Kohli J, Asher J, Gaya JB, Marcus J, Tang J, Chan J, Zhen J, Reizenstein J, Teboul J, Zhong J, Jin J, Yang J, Cummings J, Carvill J, Shepard J, McPhie J, Torres J, Ginsburg J, Wang J, Wu KK, KamHou U, Saxena K, Prasad K, Khandelwal K, Zand K, Matosich K, Veeraraghavan K, Michelena K, Li K, Huang K, Chawla K, Lakhotia K, Huang K, Chen L, Garg L, Lavender A, Silva L, Bell L, Zhang L, Guo L, Yu L, Moshkovich L, Wehrstedt L, Khabsa M, Avalani M, Bhatt M, Tsimpoukelli M, Mankus M, Hasson M, Lennie M, Reso M, Groshev M, Naumov M, Lathi M, Keneally M, Seltzer ML, Valko M, Restrepo M, Patel M, Vyatskov M, Samvelyan M, Clark M, Macey M, Wang M, Jubert Hermoso MJ, Metanat M, Rastegari M, Bansal M, Santhanam N, Parks N, White N, Bawa N, Singhal N, Egebo N, Usunier N, Laptev NP, Dong N, Zhang N, Cheng N, Chernoguz O, Hart O, Salpekar O, Kalinli O, Kent P, Parekh P, Saab P, Balaji P, Rittner P, Bontrager P, Roux P, Dollár P, Zvyagina P, Ratanchandani P, Yuvraj P, Liang Q, Alao R, Rodriguez R, Ayub R, Murthy R, Nayani R, Mitra R, Li R, Hogan R, Battey R, Wang R, Maheswari R, Howes R, Rinott R, Bondu SJ, Datta S, Chugh S, Hunt S, Dhillon S, Sidorov S, Pan S, Verma S, Yamamoto S, Ramaswamy S, Lindsay S, Feng S, Lin S, Zha SC, Shankar S, Zhang S, Wang S, Agarwal S, Sajuyigbe S, Chintala S, Max S, Chen S, Kehoe S, Satterfield S, Govindaprasad S, Gupta S, Cho SB, Virk S, Subramanian S, Choudhury S, Goldman S, Remez T, Glaser T, Best T, Kohler T, Robinson T, Li T, Zhang T, Matthews T, Chou T, Shaked T, Vontimitta V, Ajayi V, Montanez V, Mohan V, Kumar VS, Mangla V, Ionescu V, Poenaru VA, Mihailescu VT, Ivanov V, Li W, Wang W, Jiang W, Bouaziz W, Constable W, Tang X, Wang X, Wu X, Wang X, Xia X, Wu X, Gao X, Chen Y, Hu Y, Jia Y, Qi Y, Li Y, Zhang Y, Zhang Y, Adi Y, Nam Y, Wang Y, Hao Y, Qian Y, He Y, Rait Z, DeVito Z, Rosnbrick Z, Wen Z, Yang Z, Zhao Z. The Llama 3 Herd of Models. ArXiv. 2024;abs/2407.21783.

5. <u>a</u>, <u>b</u>*Yang A, Yang B, Hui B, Zheng B, Yu B, Zhou C, Li C, Li C, Liu D, Huang F, Dong G, Wei H, Lin H, Tang J, Wang J, Yang J, Tu J, Zhang J, Ma J, Xu J, Zhou J, Bai J, He J, Lin J, Dang K, Lu K, Chen K, Yang K, Li M, Xue M, Ni N, Zhang P, Wang P, Peng R, Men R, Gao R, Lin R, Wang S, Bai S, Tan S, Zhu T, Li T, Liu T, Ge W, Deng X, Zhou X, Ren X, Zhang X, Wei X, Ren X, Fan Y, Yao Y, Zhang Y, Wan Y, Chu Y, Cui Z, Zhang Z, Fan Z (2024). "Qwen2 Technical Report". ArXiv.* **abs/2407.10671.** *S2CID <u>271212307</u>.*

6. ⌃Radford A, Kim JW, Hallacy C, Ramesh A, Goh G, Agarwal S, Sastry G, Askell A, Mishkin P, Clark J, Krue ger G, Sutskever I. "Learning transferable visual models from natural language supervision." In: Interna tional Conference on Machine Learning; 2021.

7. a, bZhai X, Mustafa B, Kolesnikov A, Beyer L (2023). "Sigmoid Loss for Language Image Pre-Training". 2023 IEEE/CVF International Conference on Computer Vision (ICCV). 2023: 11941–11952.

8. ⌃Sun Q, Fang Y, Wu LY, Wang X, Cao Y (2023). "EVA-CLIP: Improved Training Techniques for CLIP at Sc ale". ArXiv. abs/2303.15389.

9. ⌃Oquab M, Darcet T, Moutakanni T, Vo HQ, Szafraniec M, Khalidov V, Fernandez P, Haziza D, Massa F, El-Nouby A, Assran M, Ballas N, Galuba W, Howes R, Huang P-Y (Bernie), Li S-W, Misra I, Rabbat MG, Sharma V, Synnaeve G, Xu H, Jégou H, Mairal J, Labatut P, Joulin A, Bojanowski P (2023). "DINOv2: Lea rning Robust Visual Features without Supervision". ArXiv. abs/2304.07193.

10. ⌃Wang Y, Li K, Li X, Yu J, He Y, Chen G, Pei B, Zheng R, Xu J, Wang Z, Shi Y, Jiang T, Li S, Zhang H, Huang Y, Qiao Y, Wang Y, Wang L (2024). "InternVideo2: Scaling Video Foundation Models for Multimodal Vid eo Understanding". ArXiv. abs/2403.15377.

11. a, b, c, dLi K, Wang Y, He Y, Li Y, Wang Y, Liu Y, Wang Z, Xu J, Chen G, Luo P, Wang L, Qiao Y. "MVBench: A Comprehensive Multi-modal Video Understanding Benchmark". 2024 IEEE/CVF Conference on Comp uter Vision and Pattern Recognition (CVPR). 2023:22195–22206.

12. a, bLiu R, Li C, Tang H, Ge Y, Shan Y, Li G (2024). "ST-LLM: Large Language Models Are Effective Temp oral Learners". ArXiv. abs/2404.00308.

13. a, bLi F, Zhang R, Zhang H, Zhang Y, Li B, Li W, Ma Z, Li C (2024). "LLaVA-NeXT-Interleave: Tackling Multi-image, Video, and 3D in Large Multimodal Models". ArXiv. abs/2407.07895. Available from: http s://arxiv.org/abs/2407.07895.

14. a, b, c, d, e, fLi B, Zhang Y, Guo D, Zhang R, Li F, Zhang H, Zhang K, Li Y, Liu Z, Li C (2024). "LLaVA-OneV ision: Easy Visual Task Transfer". ArXiv. abs/2408.03326. Available from: https://arxiv.org/abs/2408.03 326.

15. a, bZhang Y, Wu J, Li W, Li B, Ma Z, Liu Z, Li C (2024). "Video Instruction Tuning With Synthetic Data".

16. a, bWang Y, Wang Y, Wu P, Liang J, Zhao D, Zheng Z (2024). "Efficient Temporal Extrapolation of Multi modal Large Language Models with Temporal Grounding Bridge".

17. a, bKrishna R, Hata K, Ren F, Fei-Fei L, Niebles JC (2017). "Dense-Captioning Events in Videos". 2017 IE EE International Conference on Computer Vision (ICCV). pages 706–715.

18. [a, b]Gao J, Sun C, Yang Z, Nevatia R (2017). "TALL: Temporal Activity Localization via Language Query". *2017 IEEE International Conference on Computer Vision (ICCV). pages 5277-5285.*

19. [a, b, c]Hendricks LA, Wang O, Shechtman E, Sivic J, Darrell T, Russell BC (2017). "Localizing moments in video with natural language". *2017 IEEE International Conference on Computer Vision (ICCV). pages 5804-5813.*

20. [a, b, c]Lei J, Berg TL, Bansal M (2021). "QVHighlights: Detecting Moments and Highlights in Videos via Natural Language Queries". *ArXiv. **abs/2107.09609.***

21. [a, b, c]Zhou L, Xu C, Corso JJ. "Towards Automatic Learning of Procedures From Web Instructional Videos." *In: AAAI Conference on Artificial Intelligence; 2017. S2CID 19713015.*

22. [^]Xiao J, Yao A, Li Y, Chua T-S (2023). "Can I Trust Your Answer? Visually Grounded Video Question Answering". *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pages 13204-13214.*

23. [a, b]Schwartz E, Choshen L, Shtok J, Doveh S, Karlinsky L, Arbelle A (2024). "NumeroLogic: Number Encoding for Enhanced LLMs' Numerical Reasoning". *ArXiv. **abs/2404.00459.***

24. [^]Wang Y, Xie C, Liu Y, Zheng Z (2024). "VideoLLaMB: Long-context Video Understanding with Recurrent Memory Bridges". *ArXiv. **abs/2409.01071**. Available from: https://arxiv.org/abs/2409.01071.*

25. [a, b]Xu Z, Meng X, Wang Y, Su Q, Qiu Z, Jiang X, Liu Q. Learning summary-worthy visual representation for abstractive summarization in video. *In: Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23. International Joint Conferences on Artificial Intelligence Organization; 2023. p. 5242-5250. doi:10.24963/ijcai.2023/582.*

26. [^]Gu J, Meng X, Lu G, Hou L, Minzhe N, Liang X, et al. Wukong: A 100 Million Large-scale Chinese Cross-modal Pre-training Benchmark. *In: Koyejo S, Mohamed S, Agarwal A, Belgrave D, Cho K, Oh A, editors. Advances in Neural Information Processing Systems. Curran Associates, Inc.; 2022. p. 26418-26431. Available from: https://proceedings.neurips.cc/paper_files/paper/2022/file/a90b9a09a6ee43d6631cf42e225d73b4-Paper-Datasets_and_Benchmarks.pdf.*

27. [a, b, c, d]Ren S, Yao L, Li S, Sun X, Hou L. "TimeChat: A Time-sensitive Multimodal Large Language Model for Long Video Understanding". *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2023: 14313-14323.*

28. [a, b]Huang B, Wang X, Chen H, Song Z, Zhu W (2023). "VTimeLLM: Empower LLM to Grasp Video Moments". *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pages 14271-14280.*

29. [a, b]Guo Y, Liu J, Li M, Tang X, Chen X, Zhao B (2024). "VTG-LLM: Integrating Timestamp Knowledge in to Video LLMs for Enhanced Video Temporal Grounding". ArXiv. **abs/2405.13382**.

30. [^]Wang H, Xu Z, Cheng Y, Diao S, Zhou Y, Cao Y, Wang Q, Ge W, Huang L (2024). "Grounded-VideoLLM: Sharpening Fine-grained Temporal Grounding in Video Large Language Models".

31. [a, b, c]Wang Y, Meng X, Liang J, Wang Y, Liu Q, Zhao D (2024). "HawkEye: Training Video-Text LLMs for Grounding Text in Videos". ArXiv. **abs/2403.10228**.

32. [a, b, c]Chen J, Lv Z, Wu S, Lin KQ, Song C, Gao D, Liu JW, Gao Z, Mao D, Shou MZ (2024). "VideoLLM-online: Online Video Large Language Model for Streaming Video". 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pages 18407-18418.

33. [^]Grauman K, Westbury A, Byrne E, Chavis Z, Furnari A, Girdhar R, Hamburger J, Jiang H, Liu M, Liu X, Martin M, Nagarajan T, Radosavovic I, Ramakrishnan SK, Ryan F, Sharma J, Wray M, Xu M, Xu EZ, Zhao C, Bansal S, Batra D, Cartillier V, Crane S, Do T, Doulaty M, Erapalli A, Feichtenhofer C, Fragomeni A, Fu Q, Fuegen C, Gebreselasie AK, González C, Hillis JM, Huang X, Huang Y, Jia W, Khoo W, Kolár J, Kottur S, Kumar A, Landini F, Li C, Li Y, Li Z, Mangalam K, Modhugu R, Munro J, Murrell T, Nishiyasu T, Price W, Ruiz Puentes P, Ramazanova M, Sari L, Somasundaram KK, Southerland A, Sugano Y, Tao R, Vo M, Wang Y, Wu X, Yagi T, Zhu Y, Arbeláez P, Crandall DJ, Damen D, Farinella GM, Ghanem B, Ithapu VK, Jawahar CV, Joo H, Kitani K, Li H, Newcombe RA, Oliva A, Park HS, Rehg JM, Sato Y, Shi J, Shou MZ, Torralba A, Torresani L, Yan M, Malik J. Ego4D: Around the World in 3,000 Hours of Egocentric Video. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2021:18973-18990.

34. [a, b, c]Tang Y, Ding D, Rao Y, Zheng Y, Zhang D, Zhao L, Lu J, Zhou J. "COIN: A Large-scale Dataset for Comprehensive Instructional Video Analysis." In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2019.

35. [a, b, c]Han M, Yang L, Chang X, Wang H (2023). "Shot2Story20K: A New Benchmark for Comprehensive Understanding of Multi-shot Videos". ArXiv. **abs/2312.10300**. S2CID 266348773.

36. [a, b]OpenAI. Hello GPT-4o. https://openai.com/index/hello-gpt-4o/, 2024. Accessed: 2024-11-13.

37. [a, b]Zala A, Cho J, Kottur S, Chen X, Ouguz B, Mehdad Y, Bansal M. "Hierarchical Video-Moment Retrieval and Step-Captioning". 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2023:23056-23065.

38. [a, b]Oncescu A-M, Henriques JF, Liu Y, Zisserman A, Albanie S (2021). "QUERYD: A Video Dataset with High-Quality Text and Audio Narrations". ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pages 2265-2269.

39. △*Xu J, Liu X, Yan J, Cai D, Li H, Li J (2022). "Learning to Break the Loop: Analyzing and Mitigating Repet itions for Neural Text Generation". ArXiv. **abs/2206.02369**. Available from: https://arxiv.org/abs/2206.0 2369.*

40. △*Keskar NS, McCann B, Varshney LR, Xiong C, Socher R (2019). "CTRL: A Conditional Transformer Lang uage Model for Controllable Generation". ArXiv. **abs/1909.05858**.*

41. △*Hu EJ, Shen Y, Wallis P, Allen-Zhu Z, Li Y, Wang S, Wang L, Chen W (2022). "LoRA: Low-Rank Adaptat ion of Large Language Models". International Conference on Learning Representations. Available from: https://openreview.net/forum?id=nZeVKeeFYf9.*

Declarations