Enrich and Detect: Video Temporal Grounding with Multimodal LLMs (Supplementary Material)

Supplementary material contents. This supplementary document is structured as follows: Section A visualizes additional qualitative results which aim to provide further insight into ED-VTG's function and performance; Section B provides additional ablations; Section C provides additional comparison with task-specific specialist baselines for fine-tuned STG task; Section D presents more details on the pseudo-label generation process; Section E discusses the instructions used for different tasks; Section F presents some failure cases; Section G explains our hyper-parameter selection; Section H provides more details on the processing of all datasets used for training (Section H.1) and fine-tuning and evaluation (Section H.2).

A. Additional Qualitative Results

We show qualitative examples in Figure A.1, where we compare ED-VTG's predictions to the TimeChat [45] baseline and the ground truth annotations. ED-VTG is trained with MIL, therefore during inference it can choose to enrich the original query if it is incomplete, or use ts as is when it is sufficient. In Figure A.2 we show example detections of ED-VTG using the predicted enriched queries and a baseline version where a model with the same architecture is trained to always use the original queries. These examples clearly demonstrate how the enriched queries often contain relevant details that enable ED-VTG to perform more accurate temporal localization than the baseline.

B. Additional Ablation Study

We conduct additional ablation experiments on two different training augmentations for query transformations compared to our cascaded enrich and detect setup, and report zero-shot numbers with increasing amount of pre-training data, showing the scalability of ED-VTG.

Offline Query Paraphrasing. In this setup, we use a blind LLaMA 3.1 8B [7] to paraphrase and grammatically correct the input queries in the training set. Notably, the LLaMA model is text-only, and does not have access to the video, and hence can not *enrich* the queries, but just paraphrases them for better grammatical construction. During evaluation, we also augment the queries in the same fashion. As shown in Table B.1, such an augmentation techniques does not bring any notable improvement on Charades and ActivityNet datasets for STG task.

Training Paradigm	Charades-STA STG			ANet-Captions STG		
	R@0.3	R@0.5	mIoU	R@0.3	R@0.5	mIoU
Detect	51.4	31.5	33.2	50.3	30.1	34.0
Offline Paraphrasing + Detect	51.4	31.6	32.7	50.5	30.8	33.9
Offline Enrich w/o Interval Anno. + Detect	51.7	31.1	31.9	49.5	29.1	32.9
Offline Enrich + Detect	51.7	31.5	33.4	49.8	29.9	33.7
Enrich & Detect	**60.1**	**37.0**	**38.4**	**56.3**	**35.5**	**37.8**

Table B.1. **Ablation on enrichment as a training pre-processing step.** We compare the proposed enrich & detect framework with two additional augmentations using LLMs. In the "Offline Paraphrasing + Detect" setup, we use a blind LLaMA 3.1 8B [7] to paraphrase and grammatically correct the input queries. In the "Offline Enrich w/o Interval Annotation + Detect" setup, we augment the queries with LLaVA OneVision 72B [21] as preprocessing, where the model sees the video, but does not have access to the ground truth labels. We observe that the proposed enrich & detect is superior since the trained model learns to perform autonomous enrichment during evaluation, which proves that the cascaded detection paradigm is significantly different than training augmentation. Reported results are in FT w/o PT setting.

Offline Query Enrichment w/o Annotated Intervals. In this second setup, we employ a multimodal LLaVA OneVision 72B model [21] for query enrichment as a form of training augmentation. Unlike the approach in Table 8 of the main paper, we do not crop the input video to the ground-truth interval in this setup. As a result, the model often incorporates irrelevant contextual information into the query, which is not helpful for localizing the desired interval. Consequently, as shown in Table B.1, this type of augmentation negatively impacts model performance. Overall, these ablation experiments demonstrate that our proposed enrich & detect approach is fundamentally different from training augmentations using LLMs. The trained model can independently enrich queries with necessary details or choose to directly ground the input query.

Pre-training Dataset Size. Table B.2 shows the effect of increasing training data on zero-shot Charades-STA STG and NExT-GQA QG datasets. We perform best when incorporating all tasks and datasets, denoting the usefulness of unified pre-training.

Comparison of Latency. We compare the inference speed of ED-VTG with and without the interval decoder on the Charades STG benchmark in ZS setting. Using the same

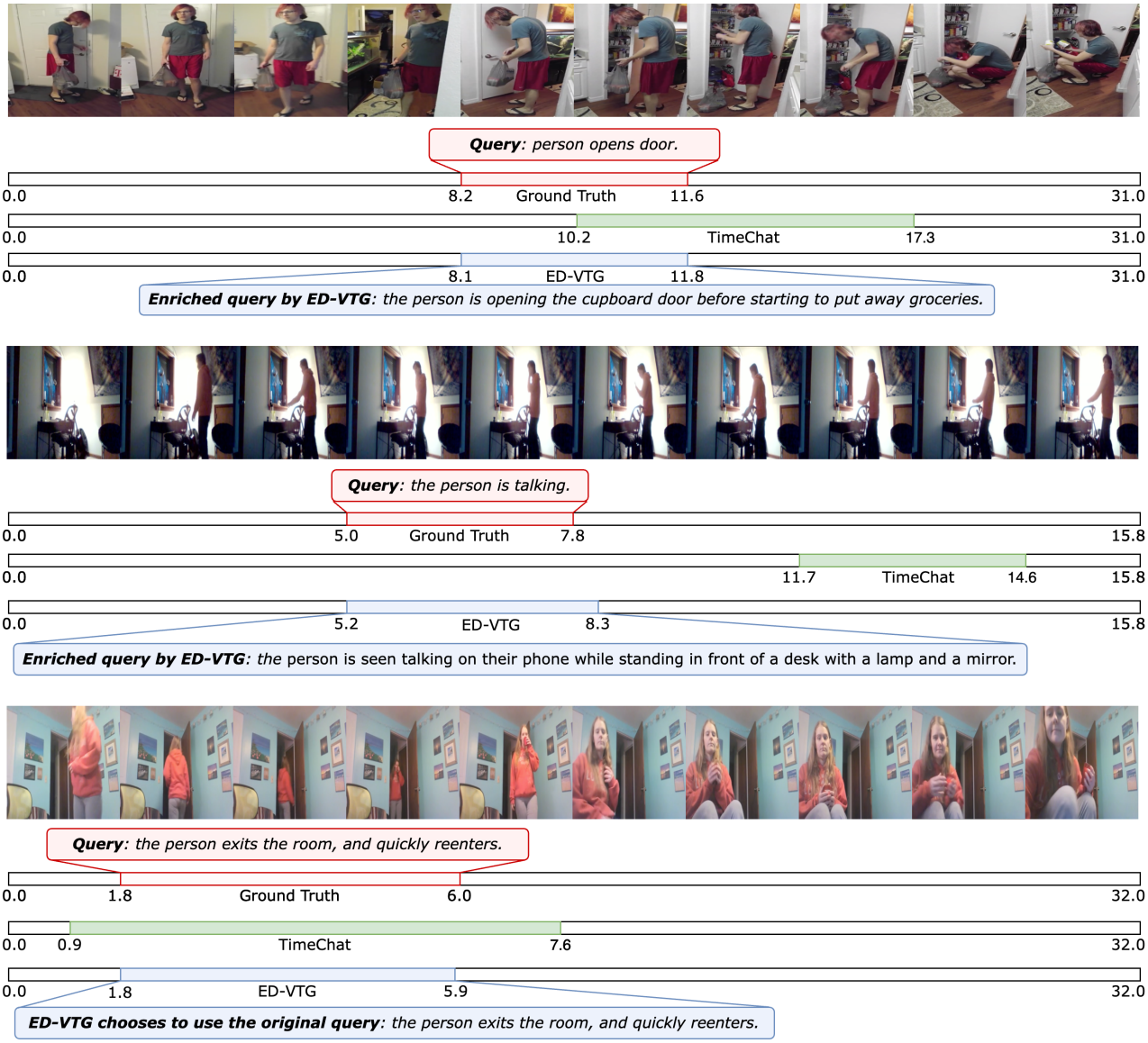


Figure A.1. **Examples of query enrichment and localization made by ED-VTG on single-query temporal grounding (STG) task from the Charades-STA [8] dataset.** We also show the prediction made by one baseline model, TimeChat [45], which directly ground the input queries using raw-text timestamp representation. Since we train ED-VTG using the MIL paradigm, the model can choose to use the input query directly or enrich it during evaluation. In the last example, since the input query is clear and explicit, the model directly localizes it.

Pre-training Tasks	# Samples	Charades-STA STG			NExT-GQA QG	
		R@0.3	R@0.5	mIoU	mIoP	mIoU
STG	91.8K	55.3	35.9	37.0	32.5	24.8
STG + VPG	133.4K	59.0	38.7	39.8	34.1	26.1
STG + VPG + AG	136K	59.5	39.1	39.9	34.2	26.6

Table B.2. **Ablation on the number of pre-training tasks and samples.** We receive the best scores when using all tasks together, showing the benefit of unified pre-training and model's scalability. Reported results are in zero-shot setting.

compute infrastructure and averaging over 3 evaluation runs, the model without decoder requires 2.10 seconds for every sample, while with decoder, it spends 2.15 seconds. Moreover, the training speeds of both models are similar, with the decoder adding only a negligible 0.2% to the total trainable parameters. This suggests that incorporating the decoder has a minimal impact on the model's latency.

Effect of interval decoder. We examine the impact of different timestamp representations in Figure B.1, comparing our lightweight decoder to using raw text or special tokens for generating time intervals. For this analysis, we fine-tune the Video-LLaMA checkpoint on the Charades and Activi-

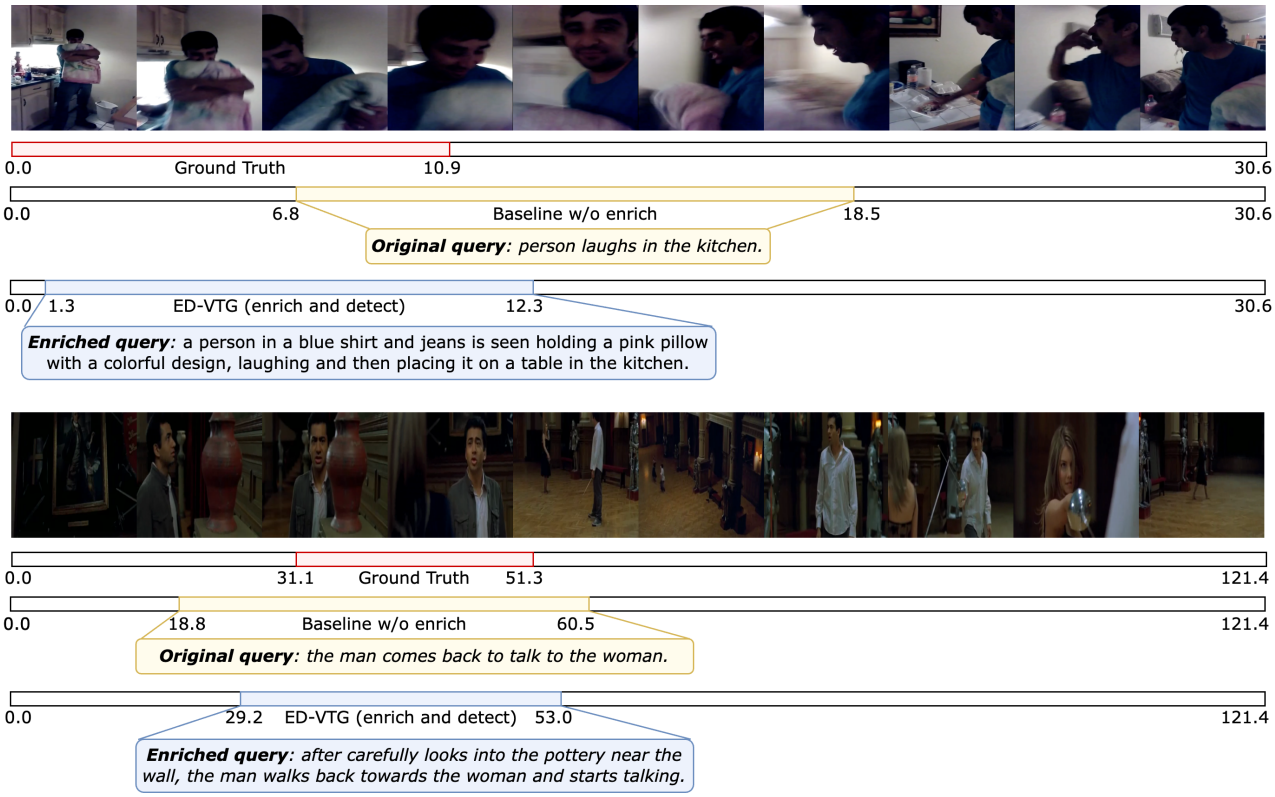
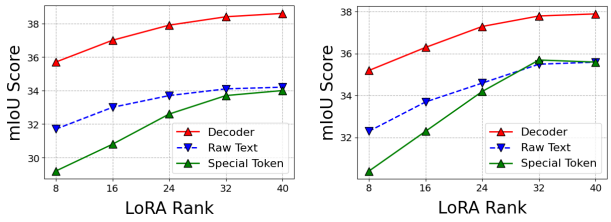


Figure A.2. **Comparison of detections of ED-VTG using its predicted enriched queries against a baseline version trained to always use the original queries.** The enriched queries contain additional relevant details and context that enable ED-VTG to perform more accurate temporal localization. In the first example, which is taken from Charades-STA [8], the additional details in the enriched query provide a more complete description of objects and actions that is more easily groundable. In the second example, sourced from the ActivityNet-Captions [18] dataset, the enriched query provides additional temporal context which leads to more precise temporal boundary prediction.



(a) Results on **Charades STG**. (b) Results on **ActivityNet STG**.

Figure B.1. **Ablation study on timestamp representation by the interval decoder.** We compare performance of our proposed lightweight decoder vs timestamp as raw text [11, 26, 34, 45] vs timestamp representation by special tokens [12, 42, 52], and find the decoder to be significantly better than both other techniques. Reported results are in FT w/o PT setting.

tyNet STG benchmarks, as shown in Figures B.1a and B.1b. Both datasets exhibit noticeable performance degradation when the decoder is omitted. Additionally, using hundreds of special tokens increases training complexity, leading to significantly poorer results at lower LoRA ranks. Since numeric digits or tokens representing frame indices lack a causal relationship in autoregressive generation, the decoder facilitates a more efficient training process. Furthermore, introducing tailored grounding objectives enables the model to produce precise timestamps.

C. Comparison with Specialist Baselines

Table C.1 extensively compares the ED-VTGwith various task-specific specialist models for the fine-tuned STG task on Charades-STA, ActivityNet-Captions, and TACoS dataset. On Charades, ED-VTG beats strong specialist baselines like UnLoc [60], UniVTG [27], MomentDiff [25], QD-DETR [37], CG-DETR [36], etc., while models like EMB [14], EaTR [15], and SG-DETR [10] perform better than ours. We observe a similar trend on the other two benchmarks. However, since the specialist models are often tailored to a particular task and dataset, they usually show poor transferability, whereas ED-VTG demonstrates state-of-the-art zero-shot performance, as shown in Table 2 of our main paper. Nevertheless, the strong performance by ED-VTG on fine-tuning setting significantly closes the gap between MLLMs and specialist baselines.

Method	Generalist Model	# Train Samples	Eval.	Charades-STA				ActivityNet-Captions				TACoS			
				R@0.3	R@0.5	R@0.7	mIoU	R@0.3	R@0.5	R@0.7	mIoU	R@0.3	R@0.5	R@0.7	mIoU
VSLNet (C3D) [65]	✗	–	FT	64.3	47.3	30.2	45.2	63.2	43.2	26.2	43.2	29.6	24.3	20.0	24.1
CTRL [9]	✗	–	FT	–	23.6	8.9	–	–	–	–	–	18.3	13.3	–	–
GTR-H [5]	✗	–	FT	–	62.6	39.7	–	–	50.6	29.1	–	–	40.4	30.2	–
2D-TAN [66]	✗	–	FT	57.3	45.8	27.9	41.1	60.3	43.4	25.0	42.5	40.0	28.0	12.9	27.2
MS-2D-TAN (I3D) [67]	✗	–	FT	–	56.6	36.2	–	62.1	45.5	28.3	–	42.0	33.6	22.1	–
Moment-DETR [20]	✗	236K	FT	65.8	52.1	30.6	45.5	–	–	–	–	38.0	24.7	12.0	25.5
UMT† [30]	✗	236K	FT	–	48.3	29.3	–	–	–	–	–	–	–	–	–
UnLoc-B [60]	✗	650K	FT	–	58.1	35.4	–	–	48.0	29.7	–	–	–	–	–
MomentDiff [25]	✗	–	FT	–	55.6	32.4	–	–	–	–	–	46.6	28.9	12.4	30.4
LGI [38]	✗	–	FT	73.0	59.5	35.5	51.4	58.5	41.5	23.1	41.1	–	–	–	–
FlashVTG (SF+C) [6]	✗	–	FT	–	60.1	38.0	–	–	–	–	–	53.7	41.8	24.7	37.6
BAM-DETR [19]	✗	–	FT	72.9	60.0	39.4	52.3	–	–	–	–	56.7	41.5	26.8	39.3
UniVTG [27]	✗	4.2M	FT	70.8	58.0	35.7	50.1	–	–	–	–	51.4	35.0	17.4	33.6
QD-DETR (SF+C) [37]	✗	–	FT	–	57.3	32.6	–	–	–	–	–	–	–	–	–
CG-DETR (SF+C) [36]	✗	–	FT	70.4	58.4	36.3	50.1	–	–	–	–	54.4	39.5	23.4	37.4
TR-DETR (SF+C) [48]	✗	–	FT	–	57.6	33.5	–	–	–	–	–	–	–	–	–
GVL (C3D) [53]	✗	–	FT	–	–	–	–	–	48.9	27.2	46.4	45.9	34.6	–	32.5
InternVideo2* + CG-DETR [55]	✗	2.1M	FT	79.7	70.0	48.9	58.8	–	–	–	–	–	–	–	–
SG-DETR [10]	✗	–	FT	–	71.1	52.8	60.7	–	–	–	–	–	46.4	33.9	42.4
MGSL-Net [29]	✗	150K	FT	–	64.0	41.0	–	–	51.9	31.4	–	42.5	32.3	–	–
EaTR [15]	✗	150K	FT	–	68.5	44.9	–	–	58.1	37.6	–	–	–	–	–
EMB (ELA) [14]	✗	–	FT	79.7	69.2	51.4	62.2	73.7	58.7	40.7	56.2	63.3	52.5	37.0	48.4
BLIP-2 (frames only) [22]	✓	129M	FT	–	43.3	32.6	–	–	25.8	9.7	–	–	–	–	–
VideoChat2 [23]	✓	2M	FT	–	–	–	–	55.5	34.7	17.7	38.9	–	–	–	–
TimeChat [45]	✓	125K	FT	–	46.7	23.7	–	–	–	–	–	27.7	15.1	6.4	18.4
HawkEye [56]	✓	715K	FT	72.5	58.3	28.8	49.3	55.9	34.7	17.9	39.1	–	–	–	–
VtimeLLM [11]	✓	170K	FT	–	–	–	–	–	–	–	–	26.8	14.4	6.1	18.0
ED-VTG	✓	136K	FT	**78.2**	**62.1**	**35.0**	**52.6**	**67.6**	**45.1**	**22.7**	**44.9**	**46.0**	**31.5**	**15.8**	**32.4**
$\Delta_{\text{Ours - HawkEye}}$	–	–	FT	5.7 ↑	3.8 ↑	6.2 ↑	3.3 ↑	11.7 ↑	10.4 ↑	4.8 ↑	5.8 ↑	–	–	–	–
$\Delta_{\text{Ours - VTimeLLM}}$	–	–	FT	–	–	–	–	–	–	–	–	19.2 ↑	17.1 ↑	9.7 ↑	14.4 ↑

Table C.1. **Extension of Table 3 in the main paper with a comprehensive list of task-specific specialist baselines.** ED-VTG beats many expert baselines, and significantly closes the gap between SOTA specialist models with MLLMs. †UMT uses video and audio as the input. *Though InterVideo2 is a generalist model, it fine-tunes CG-DETR [36] head for grounding tasks, using the LLM only as a video feature extractor.

D. Pseudo-label Generation Pipeline

Since our proposed two-step cascaded grounding approach, Enrich and Detect, requires enriched queries as ground truths during training, we augment poorly worded or potentially incomplete input queries of all training benchmarks with additional context information using an open-source and broadly capable captioning model, LLaVA OneVision (OV) 72B [21]. First, we crop the input videos between the annotated time intervals. Next, we input the original query and the cropped video to the OV model and ask it to enrich the description of the activities in the given segment while preserving the main focus of the original query. The prompt used in this step is shown in Figure D.1. To partially tackle the hallucination issue of large LLMs during language generation, next we generate a few binary choice questions from each enriched query using a text-only LLaMA 3.1 8B model [7], and filter the samples using a lower-sized OV 8B model, which is proficient at answering yes/no questions. If all descriptions in the enriched query are correct, we keep the sample; otherwise, we reiterate the process. Notably, even with our well-versed query augmentation pipeline, some enriched samples contain unimportant information for grounding, which we tackle with the proposed MIL training framework. During evaluation, we only feed the original queries as input to ED-VTG, and the model generates the enriched queries and perform grounding.

```
You are given a cropped video segment.
A brief description of the activity in this
segment is: {{Input Query}}

This activity description is written by a
human. Can you enrich the description of the
activities happening in this segment?

Make sure to preserve the meaning of the
original annotation. Enrich the query with
additional information. Moreover, keep the
enriched description brief, preferably only
one sentence.
```

Figure D.1. **Prompt for query enrichment during the pseudo-label generation using a captioning model, LLaVA OneVision 72B [21].** We feed the cropped video between the annotated time interval along with the original query, and ask the model to enrich the query with additional information while maintaining the original focus of the query.

E. Example Instructions for Different Tasks

High-quality language instructions are essential for effective instruction tuning of LLMs across various downstream

tasks [24, 41, 54]. For each task, we manually write one high-quality instruction as starting and generate variations using GPT-4 [1]. Eventually, we manually refine the LLM-generated instructions to obtain the final version. Based on insights from M^{3}IT [24] and TimeChat [45], we use six high-quality instructions per task. During training, we randomly pick one instruction for each sample. Table E.1 shows one example instruction for each task.

F. Error Analysis

Although ED-VTG learns impressive video temporal grounding capability across many different benchmarks, there are still various cases where the model fails to correctly localize the input query, especially for small and obscured objects in long videos. Moreover, since ED-VTG does not use the audio modality, acoustic expressions are sometimes hard to localize. Figure F.1 shows two such error cases. In the first example, ED-VTG fails to recognize where the person "*laughs*", primarily due to minimal relevant activities before laughter happens. As the face of the person in this video is not fully visible throughout the video, the model fails to detect such sudden and unprecedented activity. However, with acoustic information, such activities would be easy to detect. In the second case, though the query asks to localize where the "*person cracks egg*", ED-VTG produces an enriched query that contains an additional action (pouring the egg in the glass), and consequently grounds it to a longer interval. This is an example where our enrich-and-detect paradigm fails, as although the enriched query is grounded properly, this behavior is undesired. However these cases are much less common than the ones where enrichment improves the grounding, providing overall - as we have demonstrated quantitively - net performance benefit.

G. Hyper-parameter settings

Our hyper-parameter settings during the pre-training and dataset-specific fine-tuning is provided in Tables G.1 and G.2, respectively. To find the most optimal hyper-parameter combinations for different tasks and datasets, we perform a grid search on batch size, learning rate and loss weights, and report the best configuration in Table G.2.

H. Dataset Details

This section provides additional details of our pre-training, fine-tuning and evaluation datasets with an in-depth description of our pseudo-label generation pipeline.

H.1. Pre-training Datasets

DiDeMo: DiDeMo[1] [3] is a large-scale video temporal grounding dataset featuring 10,464 unique videos, annotated with natural language descriptions that highlight specific moments or events, including single-sentence summaries and shorter moment descriptions. The dataset is sourced from the Flickr Creative Commons dataset [51] and encompasses a diverse array of topics such as outdoor activities, sports, food preparation, DIY projects, travel destinations, and animals. A notable limitation of DiDeMo is that its interval annotations are made in 5-second windows, which do not capture fine-grained activities. We utilize DiDeMo for pre-training in single-query temporal grounding (STG), where the model receives an input video along with a query and is expected to output a single time interval.

QuerYD: QuerYD[2] [40], sourced from the YouDescribe project [46], is a large-scale video grounding dataset designed for moment retrieval and event localization. A distinctive feature of QuerYD is that each video includes two audio tracks: the original audio and a high-quality spoken description of the visual content. We utilize the original audio to generate automatic speech recognition (ASR) transcripts, which are then used as input for the large language model (LLM) along with task instructions. We use this dataset in the STG task format. However, since some samples in QuerYD contain single timepoint annotations instead of time intervals, we introduce a $\langle point \rangle$ token to the LLM vocabulary. During pre-training, if a $\langle point \rangle$ token is present in the ground truth, we mask out the window logit in the decoder and set the generalized intersection over union (gIoU) loss to zero.

COIN: The COIN[3] dataset [50] is a large-scale collection designed for comprehensive procedural activity recognition. It comprises over 11,800 videos covering 180 different tasks, which are organized into 12 distinct domains such as "Sports", "Leisure", "Home Improvement", "Food & Drinks" etc. Each video is meticulously annotated with step-by-step instructions, providing a detailed breakdown of the procedural activities depicted. This structure allows for the analysis of both high-level task understanding and fine-grained action recognition. The dataset is notable for its diversity, featuring videos sourced from a wide range of environments and cultural contexts, which enhances its applicability to real-world scenarios. Most important to our application, COIN includes temporal annotations that specify the start and end times of each procedural step, facilitating precise temporal action localization. We utilize COIN in the video paragraph grounding (VPG) task format, where we input multiple step descriptions as queries, and ask the

[1] https://github.com/LisaAnne/LocalizingMoments
[2] https://www.robots.ox.ac.uk/~vgg/data/queryd/
[3] https://github.com/coin-dataset/annotations

Task	Example Instructions
STG	• Please look into the given video and localize the textual query: ⟨*Input Query*⟩. If the provided query is explicit, directly localize it. Otherwise, generate an enriched version which provides more information about the desired time window without changing the main focus, and then localize it.
VPG	• Carefully review the video and textual queries provided. Your goal is to associate each query with a specific time interval in the video. If a query is clear-cut, directly localize it. For less explicit queries, develop an enhanced version that furnishes more details about the desired time window without changing the core focus, and then localize the enhanced query. Process the queries in the order they appear. The queries are: ⟨*Input Queries*⟩.
QG	• Analyze the provided video and the question: ⟨*Input Question*⟩ carefully. Your task is to identify the specific time interval in the video where the question can be accurately answered. If the question is straightforward and easily grounded, directly localize it in the video. However, if the question requires additional context or clarification, generate an enriched version that provides more information without altering its primary focus, and then determine the desired time interval.
AG	• Carefully look into the given video and the textual queries. Your job is to localize the textual queries in the video. Some of the queries may not be groundable in the input video, in that case, mention it. If a query is groundable and explicit, directly localize it. Otherwise, if the query is groundable, but lacks information, output an enriched version of the query to provide more context about the desired time window without changing the main focus, and then localize the query. Process the queries in the same order as listed in this instruction. The queries are: ⟨*Input Queries*⟩.

Table E.1. **Examples of instructions** for different tasks used by ED-VTG. Each instruction provides the model two options: (i) to perform grounding directly when the query is simple and clear, and (ii) to perform grounding in the enrich and detect paradigm, where the model first produces an enriched query with additional information about the desired time window, and then localize it. We highlight the task-specific parts in blue for every instruction.

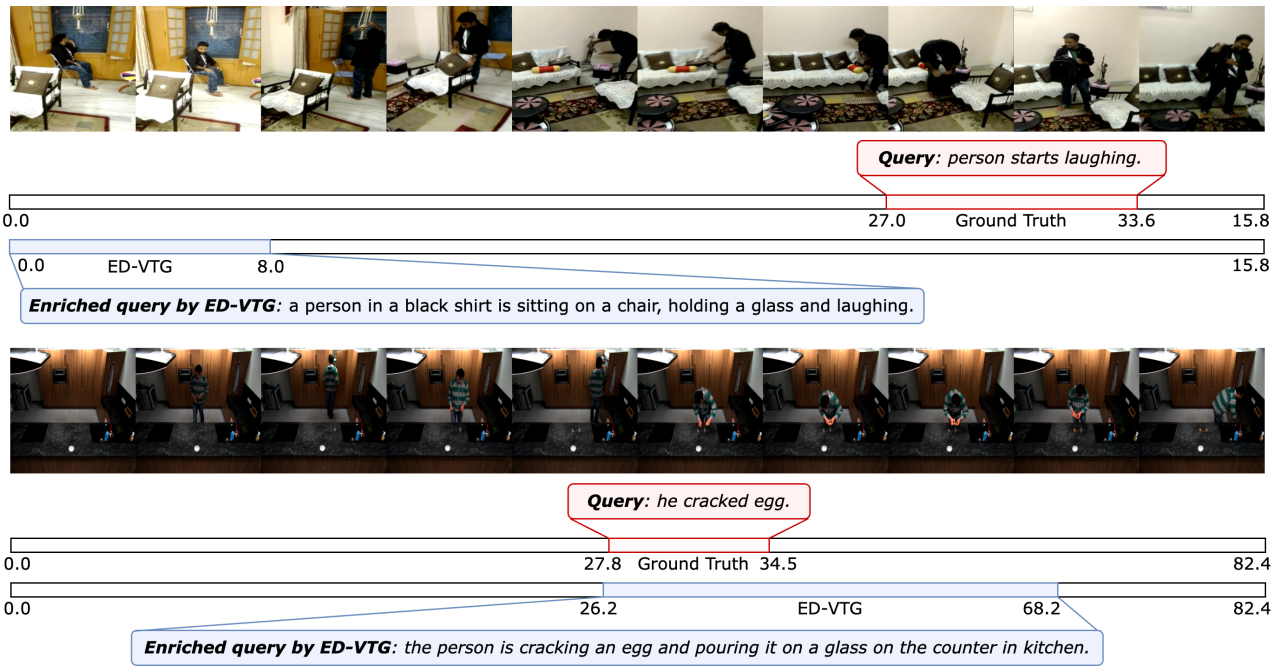


Figure F.1. **Limitations of our method.** In this figure, we show two error cases where ED-VTG fails to accurately ground the input queries. The two samples are taken from Charades-STA [8] and TACoS [44], respectively. In the first case, the model completely fails to recognize the correct interval. In the second case, ED-VTG produces an enriched query that contains an extra action compared to the original query (pouring the egg in a glass), which results in a longer temporal interval prediction which is incorrect.

Hyper-parameters	Notation	Value
Vision Encoder		
Frame encoder	–	EVA-CLIP [49]
Image Q-Former num tokens	–	32
Image Q-Former hidden layers	–	2
Video Q-Former num tokens	–	32
Video Q-Former hidden layers	–	2
Video Q-Former window size	–	32
Video Q-Former stride	–	32
Interval Decoder		
# Transformer layers	–	2
Transformer layer num heads	–	12
Transformer layer hidden dim	–	768
MLP dim	–	768 - 256 - 128 - 2
Pre-training		
Batch size	–	256
Epochs	–	40
Number of frames	–	96
Frame resolution	–	224×224
Max. length of text	–	2048
Loss weights	$\lambda_{\text{LM}}, \lambda_{\text{L1}}, \lambda_{\text{gIoU}}$	2, 1, 1
Optimizer	–	AdamW [32]
LoRA rank	–	32
Peak LR	–	$5e$-5
Warmup	–	Linear (first 8 epochs)
LR decay	–	Cosine [31]
Start LR	–	$1e$-5
End LR	–	$1e$-6
Num workers	–	6
Betas in AdamW	(β_1, β_2)	(0.9, 0.98)
Eps in AdamW	–	$1e$-8
Weight decay	–	0.05

Table G.1. **Pre-training hyper-parameter details of ED-VTG.**

model to localize each input query.

HiREST: The Hierarchical Retrieval and Step-captioning (HiREST)[4] dataset [63] supports multiple related video-text tasks within an instructional video corpus, including (1) video retrieval, (2) moment retrieval, (3) moment segmentation, and (4) step captioning. HiREST contains 1.1K high-quality, human-annotated moment spans that are relevant to text queries, making it an excellent resource for video grounding. We employ HiREST in both the single-query temporal grounding (STG) and video paragraph grounding (VPG) task formats.

VITT: The Video Timeline Tags (VITT)[5] [13] dataset provides timestamped activity descriptions for a wide range of instructional videos, focusing on hands-on skills such as cooking, car maintenance, and home repairs. It comprises approximately 8,000 videos, each averaging 7.1 segments, with each segment accompanied by a concise free-text description. While VITT is primarily used for dense video captioning, we adapt the dataset to the video paragraph grounding (VPG) format, where segment descriptions are inputted, and the system is tasked with localizing them within the video. Similar to the QuerYD dataset, samples in VITT include single timepoint annotations, for which we employ a $\langle point \rangle$ token and back-propagate using only the L1 objective.

YTTemporal: YTTemporal-1B [64] comprises 18 million narrated videos sourced from YouTube, from which we utilize the same subset as TimeChat [45]. In our approach, we employ YTTemporal in the video paragraph grounding (VPG) task setup, where the speech content from the narrations is inputted, and the model is tasked with predicting the start and end timestamps based on the video's visual signals. Due to the often poorly worded and incomplete nature of the narrations, this dataset serves as a weakly-supervised annotation source. The enriched queries significantly aid ED-VTG in achieving accurate grounding. Following the methodology of Vid2Seq [61], we use Whisper-timestamped [33, 43] to automatically transcribe the speech, which is then used as input queries.

CrossTask: The CrossTask[6] [69] dataset is a valuable resource for learning and evaluating models on cross-domain task understanding and procedural activity recognition. It consists of approximately 4,800 videos spanning 18 primary tasks and 65 related tasks, such as "Make Pancakes", "Change Car Tire" and "Assemble Shelter" each sourced from diverse domains. We use a subset of CrossTask containing 2.7K videos for article grounding (AG). Since this dataset does not contain negative queries, we generate synthetic negatives using the LLaMA 3.1 8B [7] model. We provide the model with video descriptions (dense captions and ASR) and ask it to generate negative queries that resemble the video activities but do not actually occur in the video. Afterwards, we filter the generated negative queries using multimodal LLaVA OneVision 72B [21], and manually verify a small portion (5%) of the filtered negative queries for quality assurance.

VideoCC: VideoCC[7] [39] is a large-scale dataset designed for video captioning and temporal video grounding, featuring 6.3 million video clips accompanied by 974,247 temporally-aligned captions. For our purposes, we utilize a smaller subset of 45,000 caption-interval pairs within the single-query temporal grounding (STG) task setup. The videos in this dataset span a wide array of categories, such as sports, cooking, travel, and more, offering a diverse range of scenarios for model training and evaluation. This diversity makes VideoCC an invaluable resource for developing models that can effectively understand and describe video content across various contexts. Notably, since we use only a subset of YTTemporal and VideoCC, we will easily be able to scale up our pre-training in future.

[4] https://github.com/j-min/HiREST

[5] https://github.com/google-research-datasets/Video-Timeline-Tags-ViTT

[6] https://github.com/DmZhukov/CrossTask

[7] https://github.com/google-research-datasets/videoCC-data

Task	Dataset	Fine-tuning Hyper-parameter Details									
		Batch	Epochs	Warmup	# Frames	λ_{LM}	λ_{L1}	λ_{gIoU}	Peak LR	Start LR	End LR
STG	Charades-STA [8]	32	120	24	96	2	1	1	3e-5	1e-5	1e-5
	ActivityNet-Captions [18]	32	30	6	144	1	1	1	3e-5	1e-5	1e-5
	TACoS [44]	32	120	24	144	4	1	1	3e-5	1e-5	1e-5
STG	Charades-CD-OOD [62]	32	120	24	96	2	1	1	3e-5	1e-5	1e-5
	ActivityNet-Captions [18]	32	30	6	144	3	1	1	3e-5	1e-5	1e-5
	TACoS [44]	32	120	24	144	4	1	1	3e-5	1e-5	1e-5
	YouCook2 [68]	32	120	24	144	1	1	1	3e-5	1e-5	1e-5
AG	HT-Step [2]	32	120	24	144	2	1	1	3e-5	1e-5	1e-5

Table G.2. **Fine-tuning hyper-parameter details on different datasets.** LR denotes learning rate, λ_{LM}, λ_{L1} and λ_{gIoU} denotes weights for LM, L1 and gIoU objectives, respectively. Since the NExT-GQA [59] dataset has no training split, no fine-tuning is performed on NExT-GQA, we report only zero-shot performance. All other hyper-parameters, which are not mentioned in this table, are kept the same as the pre-training setup as listed in Table G.1.

H.2. Fine-tuning and Evaluation Datasets

Charades-STA: Charades-STA[8] [8] is a specialized dataset designed for the task of temporal activity localization in videos, particularly focusing on the alignment of textual descriptions with specific video segments. Charades-STA contains 9,848 videos capturing daily indoor activities and 16,128 human-tagged query texts. Following previous works [25, 27, 37, 48], we use the train set containing 12,408 samples for fine-tuning while the test set with 3,720 samples for evaluation. We report the single-query temporal grounding (STG) results on Charades-STA.

Charades-CD-OOD: Charades-CD-OOD[9] [62] is a reorganized version of the Charades-STA dataset, specifically designed to evaluate models on their ability to generalize to out-of-distribution (OOD) scenarios in the context of paragraph grounding, which involves testing models on novel combinations of actions and objects that were not seen during training, thereby assessing their ability to extrapolate learned knowledge to new contexts. The dataset is divided into train/val/test ood sets of 4,564/333/1,440 video-paragraph pairs, respectively. The average video duration in Charades-CD-OOD is 30.60 seconds, and the average paragraph length is 2.41 sentences. We report the video paragraph grounding (VPG) performance of ED-VTG on Charades-CD-OOD.

ActivityNet-Captions: ActivityNet-Captions[10] [18] dataset is a comprehensive resource designed for dense video captioning and temporal localization tasks, derived from the original ActivityNet [18] dataset. ActivityNet-Captions features a diverse array of open-domain content, comprising 14,926 distinct videos and 19,811 localized video-paragraph pairs. On average, each video is approximately 117.63 seconds long, and each paragraph consists of about 3.63 sentences, providing detailed narrative descriptions of the video content. The dataset is structured into three subsets: training, val_1, and val_2, containing 10,009, 4,917, and 4,885 video-paragraph pairs, respectively. Consistent with prior research [4, 5, 11, 16, 28, 45, 57], we use the val_2 for evaluation. We report both STG and VPG performance of ED-VTG on ActivityNet-Captions.

TACoS: The TACoS[11] [44] dataset is a specialized collection derived from the MPII Cooking Composite Activities video corpus [47], focusing on cooking activities and kitchen scenarios. It comprises 127 videos, each accompanied by multiple paragraphs that describe the actions at varying levels of detail. Specifically, the dataset includes 1,107 video-paragraph pairs for training, 418 for validation, and 380 for testing. On average, the videos are 224.34 seconds long, and each paragraph contains approximately 8.75 sentences, providing rich and detailed descriptions of the cooking processes. The dataset's focus on cooking activities makes it an ideal benchmark for evaluating models that aim to comprehend and describe complex procedural tasks in a structured environment. We report the results on TACoS for the STG and VPG tasks.

YouCook2: The YouCook2[12] [68] dataset consists of 2,000 cooking videos sourced from YouTube, capturing a wide variety of cooking styles and cuisines from around the world. These videos are segmented into 15,400 clips, each annotated with detailed descriptions that provide step-by-step instructions for preparing various dishes. On average, each video is approximately 5.19 minutes long, and the

[8] https://github.com/jiyanggao/TALL

[9] https://github.com/yytzsy/grounding_changing_distribution/tree/main/Charades-CD

[10] http://activity-net.org/download.html

[11] https://www.mpi-inf.mpg.de/departments/computer-vision-and-machine-learning/research/vision-and-language/tacos-multi-level-corpus

[12] http://youcook2.eecs.umich.edu/download

dataset covers 89 different recipe types, offering a rich diversity of cooking scenarios. YouCook2 has 1095 and 415 ground truth video-paragraph pairs for train and evaluate, respectively. We report VPG performance of ED-VTG on YouCook2.

NExT-GQA: The NExT-GQA[13] [59] dataset is a manually annotated video question grounding dataset, where each question-answer pair is accompanied by a temporal segment annotation serving as evidence. Built upon the NExT-QA [58] dataset, NExT-GQA was created by adding 10.5K temporal labels - specifying start and end timestamps - to the QA pairs in the validation and test sets. These labels were carefully annotated and verified as crucial for understanding the questions and identifying the correct answers. Since NExT-GQA does not contain a training split, we evaluate our model's performance on zero-shot question grounding (QG) using this dataset.

HT-Step: HT-Step[14] [2] is a large-scale dataset containing temporal annotations of instructional article steps in cooking videos. It includes 116K segment-level annotations over 20K narrated videos (approximately 2.1k hours) of the HowTo100M [35] dataset. Each annotation provides a temporal interval and a categorical step label from a taxonomy of 4,958 unique steps automatically mined from wikiHow articles [17], which include rich descriptions of each step. Since HTStep releases the negative queries, we report article grounding (AG) performance on this dataset.

[13] https://github.com/doc-doc/NExT-GQA

[14] https://github.com/facebookresearch/htstep

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 5

[2] Triantafyllos Afouras, Effrosyni Mavroudi, Tushar Nagarajan, Huiyu Wang, and Lorenzo Torresani. Ht-step: Aligning instructional articles with how-to videos. In *NeurIPS*, 2024. 8, 9

[3] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *ICCV*, pages 5803–5812, 2017. 5

[4] Peijun Bao, Qian Zheng, and Yadong Mu. Dense events grounding in video. In *AAAI*, pages 920–928, 2021. 8

[5] Meng Cao, Long Chen, Mike Zheng Shou, Can Zhang, and Yuexian Zou. On pursuit of designing multi-modal transformer for video grounding. In *EMNLP*, pages 9810–9823, 2021. 4, 8

[6] Zhuo Cao, Bingqing Zhang, Heming Du, Xin Yu, Xue Li, and Sen Wang. Flashvtg: Feature layering and adaptive score handling network for video temporal grounding. In *WACV*, 2024. 4

[7] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. 1, 4, 7

[8] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. Tall: Temporal activity localization via language query. In *ICCV*, pages 5267–5275, 2017. 2, 3, 6, 8

[9] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. Tall: Temporal activity localization via language query. In *ICCV*, pages 5267–5275, 2017. 4

[10] Aleksandr Gordeev, Vladimir Dokholyan, Irina Tolstykh, and Maksim Kuprashevich. Saliency-guided detr for moment retrieval and highlight detection. *arXiv preprint arXiv:2410.01615*, 2024. 3, 4

[11] Bin Huang, Xin Wang, Hong Chen, Zihan Song, and Wenwu Zhu. Vtimellm: Empower llm to grasp video moments. In *CVPR*, pages 14271–14280, 2024. 3, 4, 8

[12] De-An Huang, Shijia Liao, Subhashree Radhakrishnan, Hongxu Yin, Pavlo Molchanov, Zhiding Yu, and Jan Kautz. Lita: Language instructed temporal-localization assistant. In *ECCV*, pages 202–218. Springer, 2025. 3

[13] Gabriel Huang, Bo Pang, Zhenhai Zhu, Clara Rivera, and Radu Soricut. Multimodal pretraining for dense video captioning. In *AACL*, pages 470–490, 2020. 7

[14] Jiabo Huang, Hailin Jin, Shaogang Gong, and Yang Liu. Video activity localisation with uncertainties in temporal boundary. In *ECCV*, pages 724–740. Springer, 2022. 3, 4

[15] Jinhyun Jang, Jungin Park, Jin Kim, Hyeongjun Kwon, and Kwanghoon Sohn. Knowing where to focus: Event-aware transformer for video grounding. In *CVPR*, pages 13846–13856, 2023. 3, 4

[16] Xun Jiang, Xing Xu, Jingran Zhang, Fumin Shen, Zuo Cao, and Heng Tao Shen. Semi-supervised video paragraph grounding with contrastive encoder. In *CVPR*, pages 2466–2475, 2022. 8

[17] Mahnaz Koupaee and William Yang Wang. Wikihow: A large scale text summarization dataset. *arXiv preprint arXiv:1810.09305*, 2018. 9

[18] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *ICCV*, pages 706–715, 2017. 3, 8

[19] Pilhyeon Lee and Hyeran Byun. Bam-detr: Boundary-aligned moment detection transformer for temporal sentence grounding in videos. In *ECCV*, pages 220–238. Springer, 2024. 4

[20] Jie Lei, Tamara L Berg, and Mohit Bansal. Detecting moments and highlights in videos via natural language queries. *NeurIPS*, 34:11846–11858, 2021. 4

[21] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *TMLR*, 2025. 1, 4, 7

[22] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, pages 19730–19742. PMLR, 2023. 4

[23] Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, et al. Mvbench: A comprehensive multi-modal video understanding benchmark. In *CVPR*, pages 22195–22206, 2024. 4

[24] Lei Li, Yuwei Yin, Shicheng Li, Liang Chen, Peiyi Wang, Shuhuai Ren, Mukai Li, Yazheng Yang, Jingjing Xu, Xu Sun, et al. M^3IT: A Large-Scale Dataset towards Multi-Modal Multilingual Instruction Tuning. *arXiv preprint arXiv:2306.04387*, 2023. 5

[25] Pandeng Li, Chen-Wei Xie, Hongtao Xie, Liming Zhao, Lei Zhang, Yun Zheng, Deli Zhao, and Yongdong Zhang. Momentdiff: Generative video moment retrieval from random to real. *NeurIPS*, 36, 2024. 3, 4, 8

[26] Zhaowei Li, Qi Xu, Dong Zhang, Hang Song, Yiqing Cai, Qi Qi, Ran Zhou, Junting Pan, Zefeng Li, Vu Tu, et al. Groundinggpt: Language enhanced multi-modal grounding model. In *ACL*, pages 6657–6678, 2024. 3

[27] Kevin Qinghong Lin, Pengchuan Zhang, Joya Chen, Shraman Pramanick, Difei Gao, Alex Jinpeng Wang, Rui Yan, and Mike Zheng Shou. Univtg: Towards unified video-language temporal grounding. In *ICCV*, pages 2794–2804, 2023. 3, 4, 8

[28] Daizong Liu, Xiaoye Qu, Jianfeng Dong, Pan Zhou, Yu Cheng, Wei Wei, Zichuan Xu, and Yulai Xie. Context-aware biaffine localizing network for temporal sentence grounding. In *CVPR*, pages 11235–11244, 2021. 8

[29] Daizong Liu, Xiaoye Qu, Xing Di, Yu Cheng, Zichuan Xu, and Pan Zhou. Memory-guided semantic learning network for temporal sentence grounding. In *AAAI*, pages 1665–1673, 2022. 4

[30] Ye Liu, Siyuan Li, Yang Wu, Chang-Wen Chen, Ying Shan, and Xiaohu Qie. Umt: Unified multi-modal transformers for joint video moment retrieval and highlight detection. In *CVPR*, pages 3042–3051, 2022. 4

[31] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. In *ICLR*, 2017. 7

[32] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 7

[33] Jérôme Louradour. whisper-timestamped., 2023. 7

[34] Kaijing Ma, Xianghao Zang, Zerun Feng, Han Fang, Chao Ban, Yuhan Wei, Zhongjiang He, Yongxiang Li, and Hao Sun. Llavilo: Boosting video moment retrieval via adapter-based multimodal modeling. In *ICCV Workshops*, pages 2790–2795. IEEE, 2023. 3

[35] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *ICCV*, pages 2630–2640, 2019. 9

[36] WonJun Moon, Sangeek Hyun, SuBeen Lee, and Jae-Pil Heo. Correlation-guided query-dependency calibration for video temporal grounding. *arXiv preprint arXiv:2311.08835*, 2023. 3, 4

[37] WonJun Moon, Sangeek Hyun, SangUk Park, Dongchan Park, and Jae-Pil Heo. Query-dependent video representation for moment retrieval and highlight detection. In *CVPR*, pages 23023–23033, 2023. 3, 4, 8

[38] Jonghwan Mun, Minsu Cho, and Bohyung Han. Local-global video-text interactions for temporal grounding. In *CVPR*, pages 10810–10819, 2020. 4

[39] Arsha Nagrani, Paul Hongsuck Seo, Bryan Seybold, Anja Hauth, Santiago Manen, Chen Sun, and Cordelia Schmid. Learning audio-video modalities from image captions. In *ECCV*, pages 407–426. Springer, 2022. 7

[40] Andreea-Maria Oncescu, Joao F Henriques, Yang Liu, Andrew Zisserman, and Samuel Albanie. Queryd: A video dataset with high-quality text and audio narrations. In *ICASSP*, pages 2265–2269. IEEE, 2021. 5

[41] Shraman Pramanick, Guangxing Han, Rui Hou, Sayan Nag, Ser-Nam Lim, Nicolas Ballas, Qifan Wang, Rama Chellappa, and Amjad Almahairi. Jack of all tasks master of many: Designing general-purpose coarse-to-fine vision-language model. In *CVPR*, pages 14076–14088, 2024. 5

[42] Long Qian, Juncheng Li, Yu Wu, Yaobo Ye, Hao Fei, Tat-Seng Chua, Yueting Zhuang, and Siliang Tang. Momentor: Advancing video large language model with fine-grained temporal reasoning. In *ICML*, 2024. 3

[43] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *ICML*, pages 28492–28518. PMLR, 2023. 7

[44] Michaela Regneri, Marcus Rohrbach, Dominikus Wetzel, Stefan Thater, Bernt Schiele, and Manfred Pinkal. Grounding action descriptions in videos. *TACL*, 1:25–36, 2013. 6, 8

[45] Shuhuai Ren, Linli Yao, Shicheng Li, Xu Sun, and Lu Hou. Timechat: A time-sensitive multimodal large language model for long video understanding. In *CVPR*, pages 14313–14323, 2024. 1, 2, 3, 4, 5, 7, 8

[46] Video Description Research and Development Center. Youdescribe, 2013. 5

[47] Marcus Rohrbach, Michaela Regneri, Mykhaylo Andriluka, Sikandar Amin, Manfred Pinkal, and Bernt Schiele. Script data for attribute-based recognition of composite activities. In *ECCV*, pages 144–157. Springer, 2012. 8

[48] Hao Sun, Mingyao Zhou, Wenjing Chen, and Wei Xie. Tr-detr: Task-reciprocal transformer for joint moment retrieval and highlight detection. In *AAAI*, pages 4998–5007, 2024. 4, 8

[49] Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023. 7

[50] Yansong Tang, Dajun Ding, Yongming Rao, Yu Zheng, Danyang Zhang, Lili Zhao, Jiwen Lu, and Jie Zhou. Coin: A large-scale dataset for comprehensive instructional video analysis. In *CVPR*, pages 1207–1216, 2019. 5

[51] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. Yfcc100m: The new data in multimedia research. *Communications of the ACM*, 59(2):64–73, 2016. 5

[52] Haibo Wang, Zhiyang Xu, Yu Cheng, Shizhe Diao, Yufan Zhou, Yixin Cao, Qifan Wang, Weifeng Ge, and Lifu Huang. Grounded-videollm: Sharpening fine-grained temporal grounding in video large language models. *arXiv preprint arXiv:2410.03290*, 2024. 3

[53] Teng Wang, Jinrui Zhang, Feng Zheng, Wenhao Jiang, Ran Cheng, and Ping Luo. Learning grounded vision-language representation for versatile understanding in untrimmed videos. *arXiv preprint arXiv:2303.06378*, 2023. 4

[54] Wenhai Wang, Zhe Chen, Xiaokang Chen, Jiannan Wu, Xizhou Zhu, Gang Zeng, Ping Luo, Tong Lu, Jie Zhou, Yu Qiao, et al. Visionllm: Large language model is also an open-ended decoder for vision-centric tasks. *NeurIPS*, 36, 2024. 5

[55] Yi Wang, Kunchang Li, Xinhao Li, Jiashuo Yu, Yinan He, Chenting Wang, Guo Chen, Baoqi Pei, Rongkun Zheng, Jilan Xu, Zun Wang, et al. Internvideo2: Scaling video foundation models for multimodal video understanding. *arXiv preprint arXiv:2403.15377*, 2024. 4

[56] Yueqian Wang, Xiaojun Meng, Jianxin Liang, Yuxuan Wang, Qun Liu, and Dongyan Zhao. Hawkeye: Training video-text llms for grounding text in videos. *arXiv preprint arXiv:2403.10228*, 2024. 4

[57] Zhenzhi Wang, Limin Wang, Tao Wu, Tianhao Li, and Gangshan Wu. Negative sample matters: A renaissance of metric learning for temporal grounding. In *AAAI*, pages 2613–2623, 2022. 8

[58] Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. Next-qa: Next phase of question-answering to explaining temporal actions. In *CVPR*, pages 9777–9786, 2021. 9

[59] Junbin Xiao, Angela Yao, Yicong Li, and Tat-Seng Chua. Can i trust your answer? visually grounded video question answering. In *CVPR*, pages 13204–13214, 2024. 8, 9

[60] Shen Yan, Xuehan Xiong, Arsha Nagrani, Anurag Arnab, Zhonghao Wang, Weina Ge, David Ross, and Cordelia Schmid. Unloc: A unified framework for video localization tasks. In *CVPR*, pages 13623–13633, 2023. 3, 4

[61] Antoine Yang, Arsha Nagrani, Paul Hongsuck Seo, Antoine Miech, Jordi Pont-Tuset, Ivan Laptev, Josef Sivic, and Cordelia Schmid. Vid2seq: Large-scale pretraining of a visual language model for dense video captioning. In *CVPR*, pages 10714–10726, 2023. 7

[62] Yitian Yuan, Xiaohan Lan, Xin Wang, Long Chen, Zhi Wang, and Wenwu Zhu. A closer look at temporal sentence grounding in videos: Dataset and metric. In *Proceedings of the 2nd international workshop on human-centric multimedia analysis*, pages 13–21, 2021. 8

[63] Abhay Zala, Jaemin Cho, Satwik Kottur, Xilun Chen, Barlas Oguz, Yashar Mehdad, and Mohit Bansal. Hierarchical video-moment retrieval and step-captioning. In *CVPR*, pages 23056–23065, 2023. 7

[64] Rowan Zellers, Jiasen Lu, Ximing Lu, Youngjae Yu, Yanpeng Zhao, Mohammadreza Salehi, Aditya Kusupati, Jack Hessel, Ali Farhadi, and Yejin Choi. Merlot reserve: Neural script knowledge through vision and language and sound. In *CVPR*, pages 16375–16387, 2022. 7

[65] Hao Zhang, Aixin Sun, Wei Jing, and Joey Tianyi Zhou. Span-based localizing network for natural language video localization. In *ACL*, pages 6543–6554, 2020. 4

[66] Songyang Zhang, Houwen Peng, Jianlong Fu, and Jiebo Luo. Learning 2d temporal adjacent networks for moment localization with natural language. In *AAAI*, pages 12870–12877, 2020. 4

[67] Songyang Zhang, Houwen Peng, Jianlong Fu, Yijuan Lu, and Jiebo Luo. Multi-scale 2d temporal adjacency networks for moment localization with natural language. *IEEE TPAMI*, 44(12):9073–9087, 2021. 4

[68] Luowei Zhou, Chenliang Xu, and Jason Corso. Towards automatic learning of procedures from web instructional videos. In *AAAI*, 2018. 8

[69] Dimitri Zhukov, Jean-Baptiste Alayrac, Ramazan Gokberk Cinbis, David Fouhey, Ivan Laptev, and Josef Sivic. Cross-task weakly supervised learning from instructional videos. In *CVPR*, pages 3537–3545, 2019. 7