

Enrich and Detect: Video Temporal Grounding with Multimodal LLMs

Shraman Pramanick^{◦1,2}[✉] Effrosyni Mavroudi¹ Yale Song¹ Rama Chellappa²
Lorenzo Torresani³ Triantafyllos Afouras¹[✉]

¹FAIR, Meta, ²Johns Hopkins University ³Northeastern University

<https://shramanpramanick.github.io/ED-VTG/>

Abstract

We introduce ED-VTG, a method for fine-grained video temporal grounding utilizing multi-modal large language models. Our approach harnesses the capabilities of multimodal LLMs to jointly process text and video, in order to effectively localize natural language queries in videos through a two-stage process. Rather than being directly grounded, language queries are initially transformed into enriched sentences that incorporate missing details and cues to aid in grounding. In the second stage, these enriched queries are grounded, using a lightweight decoder, which specializes at predicting accurate boundaries conditioned on contextualized representations of the enriched queries. To mitigate noise and reduce the impact of hallucinations, our model is trained with a multiple-instance-learning objective that dynamically selects the optimal version of the query for each training sample. We demonstrate state-of-the-art results across various benchmarks in temporal video grounding and paragraph grounding settings. Experiments reveal that our method significantly outperforms all previously proposed LLM-based temporal grounding approaches and is either superior or comparable to specialized models, while maintaining a clear advantage against them in zero-shot evaluation scenarios.

1. Introduction

Video temporal grounding [38, 47, 63, 92, 119] aims to identify temporal intervals in a video that correspond to a set of provided language queries. The task is essential for applications such as video editing and content retrieval. Conversely, video captioning [1, 69, 90, 91, 121] entails generating a natural language description for a given video segment, effectively translating visual content into text. The two tasks are in fact dual, as the outputs of one task are the inputs to the other, and vice versa. Intuitively, there is significant potential in exploiting this synergy, however it has largely remained unexplored, with previous works typically

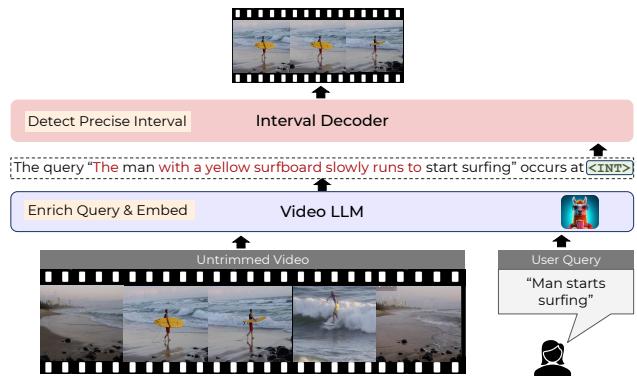


Figure 1. **Our proposed system.** ED-VTG performs video temporal grounding as a two-stage process: since user queries are often incomplete or coarse, the first stage involves producing an enriched query that adds additional details to the original, making it easier to ground. Meanwhile, a contextualized embedding is generated, containing all the information about the interval to be predicted. In the second stage, an interval decoder translates these embeddings into precise temporal boundaries.

specializing in one of the two tasks [11, 38, 63, 121] or solving them in multi-task setting [82, 98] without investigating how each task can benefit the other.

In this work, we exploit this duality by leveraging captioning to enhance grounding. Our key observation is that natural language queries often lack the completeness or detail necessary for effective temporal localization; indeed, existing grounding datasets frequently contain poorly worded, coarse, and potentially incomplete queries. The quality and completeness of these queries however is crucial for the precision of the temporal grounding. A natural hypothesis, therefore, is that more detailed queries, which could be obtained via conditional captioning, can significantly enhance grounding accuracy. For example, as illustrated in Figure 1, a vague query like ‘*Man starts surfing*’ can be refined into a more detailed description such as ‘*The man with a yellow surfboard slowly runs to start surfing*,’ resulting in grounding with more accurate temporal boundaries. In other cases, refining an abstract concept may in-

*Work done during an internship at FAIR.

[✉]spraman3@jhu.edu, afourast@meta.com

volve breaking down a complex query into simpler components that are more directly groundable.

This key idea of query enrichment forms the basis of our proposed approach. Concretely, we transform grounding into a two-stage reasoning process using a multi-modal LLM: first the model enriches the input query into a more detailed description by adding missing details based on the video content, and then temporally localizes the resulting enriched query in the video.

To effectively perform the temporal localization, we introduce a lightweight perception decoder that specializes in generating precise temporal boundaries, conditioned on a contextual representation, allowing the LLM to focus on language outputs, where it excels. The perception decoder allows us to leverage purposefully crafted training objectives for temporal grounding, building on prior knowledge and task-specific characteristics developed from years of research in object detection [22, 24, 81, 89] and temporal localization [38, 47, 63, 119, 122].

Learning to jointly *enrich and detect* requires high-quality enriched query labels. We obtain those by using a strong external captioning model which we condition on the original queries and the video content of the target temporal boundaries. However powerful, these models are prone to hallucinations and there is no guarantee that the enriched queries will always be easier to ground than the original ones. At the same time, annotating the ground truth to determine which query – original or enriched – is a better candidate, is extremely expensive. To address this issue, we propose training in a multiple-instance learning (MIL) fashion, that enables the model to autonomously determine which query is better suited for the task during training.

Finally, we note that our proposed method is not equivalent to a data augmentation approach which simply preoccesses the training set to generate enhanced queries that are directly used for training. While this simpler alternative offers some of the same benefits, it suffers from the limitation that extracting enriched queries during training requires knowledge of the ground-truth temporal segments, *i.e.*, the grounding targets. Since during inference these segments are unknown, the original queries must be used as input, which, as we will show experimentally, is suboptimal. Our method overcomes this limitation by learning to jointly enrich and detect, demonstrating superior performance.

To summarize, our contributions are as follows: (i) we introduce a cascaded approach to temporal grounding, where the model first enriches the provided language query based on the video context and then proceeds to localize it; (ii) we enable multi-modal LLMs to accurately localize text queries using a lightweight decoder which allows for training with detection objectives tailored to the task; (iii) we propose a multiple-instance learning paradigm that enables the model to dynamically select the query that leads to

better temporal localization; (iv) we achieve state-of-the-art results on several temporal grounding benchmarks, for both single query grounding and paragraph grounding, demonstrating, for the first time, an LLM-based model that surpasses or performs comparably to specialist models.

2. Related Works

LLM-based temporal grounding. Prior works have explored using LLMs for grounding natural language sentences in videos, either using raw text tokens to represent timestamps [26, 42, 46, 57, 82, 116] or by adding hundreds of special tokens to the LLM’s vocabulary to represent video frames [27, 73, 98]. Our approach differs from these methods in that by utilizing a lightweight interval decoder we can apply detection losses such as L1 and gIoU with minimal added complexity; we additionally exploit the LLM’s potential to describe video content in detail.

Specialist models. There is a rich variety of specialist models in the literature that are tailored to specific variants of temporal grounding, *e.g.* single query temporal grounding [119] and video paragraph grounding [94], and as such, achieve strong performance [7, 32, 48, 55, 86, 94, 115, 122, 123]. Modern methods typically employ a multi-modal transformer [119] that fuses dense video features with text embeddings of the language query, followed by a specialized detector head for performing detections [21, 38, 63, 85–87]. However, because these models are often trained on limited datasets for such a narrowly defined task, they struggle with generalization. Indeed, the zero-shot performance of these methods is limited [17, 106, 107, 109, 111]. In this work, we aim to address the shortcomings of previous methods by fully combining the generalization abilities of multi-modal LLMs with the advantages of specialist models.

Dense captioning. Our method is related to dense video captioning [36], where the objective is to segment a given video into multiple parts and simultaneously provide descriptive captions for each segment. Traditional approaches tackle this task either by first determining the segments and then providing descriptions [7, 26, 30, 45, 128, 130], or jointly learning both tasks [13, 45, 110, 129, 133]. Recent advances have demonstrated video-conditioned LLMs to excel in this task [104, 132, 134]. While there are similarities between our enrich-and-detect paradigm and dense captioning, we solve a different task, namely video temporal grounding, where the input query is given and constrains the problem.

Prompt augmentation with LLMs. Off-the-shelf LLMs have recently been used to augment input prompts for tasks such as image retrieval and image classification by providing additional, clarifying descriptions. These augmented descriptions can aid generalization in various vision and NLP tasks such as visual question answering [15, 72, 96],

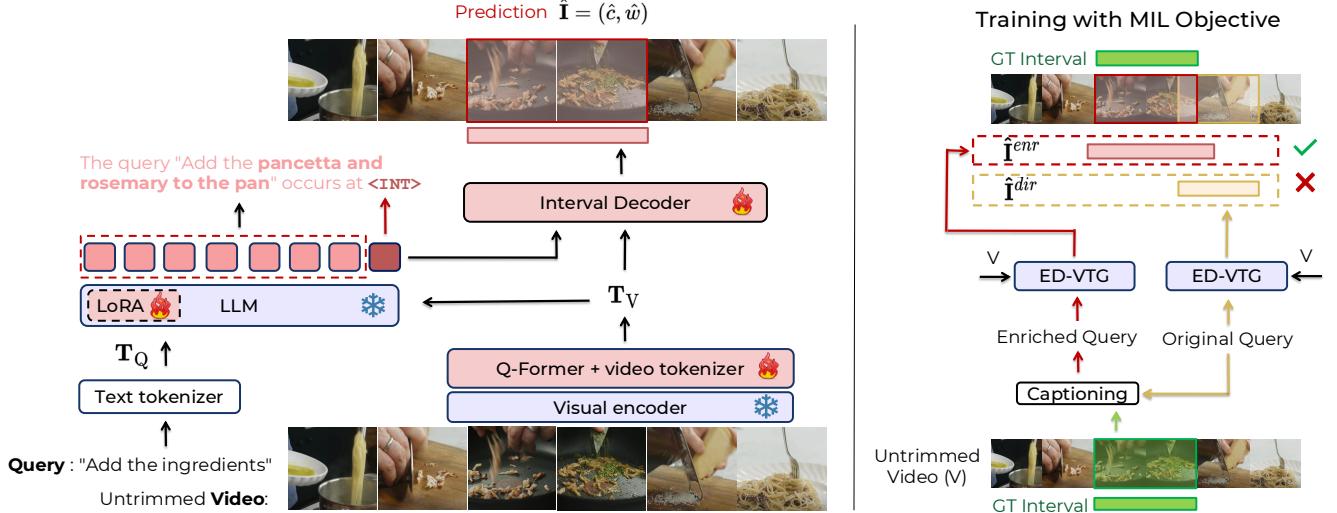


Figure 2. Overview of the proposed Enrich and Detect framework: (left) ED-VTG pipeline: Given an untrimmed video and a query Q to be grounded, the inputs are first tokenized into video tokens T_V and text tokens T_Q . The tokens are then fed into an LLM, which first generates an enriched query by, e.g., filling in any missing details and then emitting an interval token $<\text{INT}>$. The embedding of this special token is finally decoded into the predicted temporal interval via a lightweight interval decoder. In the example shown here, the vague input query is enriched into a more detailed one by our model which can be subsequently grounded more easily. (right) Training: ED-VTG is trained using ground-truth temporal intervals and pseudo-labels of enriched queries, generated by an external off-the-shelf captioning model [39], which – unlike our model at inference time – has access to the ground-truth intervals. For every sample during training, the proposed multiple instance learning (MIL) framework allows ED-VTG to assess both the original or the enriched queries and generate two sets of predictions, \hat{I}^{enr} , the interval predicted using the enriched query, and \hat{I}^{dir} using the original query. Next, the model backpropagates using the better prediction (i.e., lower grounding loss). Hence, during training, ED-VTG dynamically learns to decide for which sample enrichment is necessary and, based on that, performs detection.

dialog generation [20], and visual classification [60]. Our approach goes beyond using an off-the-shelf LLM for offline query enrichment, rather learning to dynamically enrich queries during inference time towards better grounding.

Prior work has also investigated the use of captioners and language models for data augmentation by rephrasing existing descriptions or providing new improved ones. LaViLa[126] uses rephraser and narrator models to improve the quality of the training data for video-text alignment. Augmentation during training is sufficient for that task, as the goal is to learn joint representations; for our grounding task however simply augmenting the training set, without enriching queries during inference is not as effective, as we empirically demonstrate.

Multiple instance learning. Multiple instance learning (MIL)[14] is a technique commonly used in weakly-supervised vision problems, when a collection of potential solutions is available but exact annotations are not. It has been successfully applied to a range of tasks, including classification [4], weakly supervised object detection[8, 12] and temporal action localization[67, 70].

3. Method

Given an untrimmed video V and a set of N associated textual queries $\mathcal{Q} = \{Q_1, \dots, Q_N\}$, temporal grounding aims to identify the corresponding temporal interval for each

query. Formally, the output is a set of temporal intervals $\mathcal{I} = \{I_1, \dots, I_N\}$. For $N = 1$, the task takes the form of single-query temporal grounding, which we will simply refer to as STG. For clarity, we describe our approach in the STG setting, without loss of generality; the pipeline however readily extends to $N > 1$.

Our approach aims to tackle temporal grounding by leveraging a multimodal LLM that (i) transforms input queries into intermediate, enriched queries by adding missing details based on the video input, (ii) generates contextualized embeddings of each latent segment to enable temporal interval prediction, (iii) decodes these embeddings into concrete temporal boundaries.

In the following, we formally introduce our proposed ED-VTG model in Section 3.1, and proceed to describe how to train it with enriched query pseudo-labels within a MIL framework (Section 3.2).

3.1. Model

Our model (Figure 2) consists of three key modules: a vision encoder that extracts video representations, a LLM that jointly processes video and language, and a lightweight interval decoder that generates precise temporal boundaries.

Enrich. Given a single input query Q about a video V with T frames, the vision encoder represents the video as a sequence of R visual tokens $T_V \in \mathbb{R}^{R \times D}$, where D is

the token dimension. The tokenized video features \mathbf{T}_V are fed along with the tokenized query \mathbf{T}_Q to the LLM, which generates an enriched query $\hat{Q}^{enr} = \{\hat{y}_1, \dots, \hat{y}_l, \dots \hat{y}_{L^{enr}}\}$ one token at a time:

$$\hat{y}_l = \mathcal{F}_{LLM}(\hat{y}_{<l}, \mathbf{T}_V, \mathbf{T}_Q). \quad (1)$$

When the model is ready to ground the query, the LLM emits a new, special token `<INT>` to trigger interval prediction. In other words, the text prediction takes the form:

$$\hat{y} = \text{"The query } \hat{Q}^{enr} \text{ occurs at } <\text{INT}>" \quad (2)$$

Detect. To detect the temporal interval corresponding to the enriched query \hat{Q}^{enr} , we introduce an interval decoder \mathcal{F}_{dec} that predicts an interval $\hat{\mathbf{I}}$ parametrized by the center \hat{c} and width \hat{w} of the predicted interval. We selected this parameterization for its advantage in decoupling position from scale, as supported by literature in object detection [50, 78, 79, 131]. The interval decoder takes the form

$$\hat{\mathbf{I}} = (\hat{c}, \hat{w}) = \mathcal{F}_{dec}(\mathcal{G}(\mathbf{h}_{int}), \mathbf{T}_V), \quad (3)$$

where $\mathbf{h}_{int} \in \mathbb{R}^D$ is the hidden state of the LLM corresponding to the `<INT>` token and \mathcal{G} is a linear projection layer. The decoder, functioning as the regression component of a temporal detector, consists of two transformer layers followed by a multi-layer perceptron (MLP), which processes a concatenation of its two inputs, $(\mathcal{G}(\mathbf{h}_{int}), \mathbf{T}_V)$. It finally outputs the predicted interval $\hat{\mathbf{I}}$, grounding the input query Q .

Notice that our formulation allows the quality of the enriched query \hat{Q}^{enr} to directly influence the accuracy of the predicted interval $\hat{\mathbf{I}}$. This is because the generation of the interval token `<INT>` (and consequently its hidden state \mathbf{h}_{int}) is conditioned on the enriched query prediction. Establishing this *cascaded* dependency chain, i.e., $(V, Q) \rightarrow \hat{Q}^{enr}$ and $(V, \hat{Q}^{enr}) \rightarrow \hat{\mathbf{I}}$, is the key idea of our approach.

3.2. Training

The model is trained end-to-end using two primary loss functions: a language modeling loss \mathcal{L}_{LM} and a temporal grounding loss \mathcal{L}_{grnd} that help supervise the “enrich” and “detect” aspects of our model, respectively.

Enrich. Given the target output text $\mathbf{y} = \{y_1, y_2, \dots, y_L\}$, \mathcal{L}_{LM} is calculated as the cross-entropy loss that evaluates the likelihood of \mathbf{y} under the predicted probability distribution generated by the model:

$$\mathcal{L}_{LM} = - \sum_{t=1}^T \log P(y_t | \mathbf{y}_{<t}, \mathbf{T}_V, \mathbf{T}_Q) \quad (4)$$

To provide a proper supervisory signal for query enrichment, we need the ground-truth pair of (Q, Q^{enr}) . However, such datasets do not exist, nor is it practical to annotate

Dataset	Domain	Tasks	Corpus	Eval. Protocol	# Train Samples	Avg Vid Length (s)	Avg Span Length (s)
DiDeMo [5]	Open	STG	PT	—	32.8K	54.57	6.49 (11.9%)
QuerYD [68]	Cooking	STG	PT	—	13.6K	158.78	7.68 (4.8%)
COIN [95]	Open	VPG	PT	—	7.5K	143.71	15.06 (10.5%)
HiREST [113]	Open	STG, VPG	PT	—	0.8K	208.03	44.50 (2.14%)
VITT [†] [28]	Open	VPG	PT	—	4.9K	287.17	—
YTTemporal [114]	Open	VPG	PT	—	28.8K	327.36	4.0 (1.2%)
CrossTask [135]	Procedural	AG	PT	—	2.7K	297.0	9.61 (3.2%)
VideoCC [65]	Open	STG	PT	—	45.0K	415.89	9.88 (2.3%)
Charades-STA [18]	Indoor	STG	FT, Eval	ZS, FT	12.4K	31.17	8.29 (26.6%)
Charades-CD-OOD [112]	Indoor	VPG	FT, Eval	FT	4.5K	30.60	7.90 (25.8%)
ANet-Captions [36]	Open	STG, VPG	FT, Eval	ZS, FT	9.5K	117.63	35.61 (30.3%)
TACoS [80]	Cooking	STG, VPG	FT, Eval	ZS, FT	9.8K	224.34	23.33 (10.4%)
YouCook2 [127]	Cooking	VPG	FT, Eval	FT	1.2K	311.41	20.07 (6.4%)
NExT-GQA [°] [107]	Open	QG	Eval	ZS	—	39.60	6.69 (16.9%)
HT-Step [3]	Cooking	AG	FT, Eval	FT	17.4K	393.89	14.88 (3.7%)

Table 1. **Dataset statistics, corresponding tasks, and evaluation protocol.** The upper side of the table represents datasets used for pre-training, resulting in a total of 136K samples. The lower side represents datasets used for fine-tuning and evaluation. We cover four different video grounding tasks: single-query temporal grounding (STG), video paragraph grounding (VPG), question grounding (QG), and article grounding (AG). QG is used only for evaluation to assess the model’s generalization capability. Average interval lengths compared to the corresponding video durations are shown in brown, denoting the annotation granularity. [†]VITT contains single timestamp annotation instead of intervals. [°]NExT-GQA contains only evaluation split.

a dataset solely for this purpose. Here, we capitalize on the tremendous amount of progress made in the video captioning literature [1, 90, 91], and use an off-the-shelf captioning model [39] to generate pseudo ground-truth Q^{enr} by refining the original query Q given its video V (see more details in Section 4.1).

Detect. Given a target temporal interval $I = (c, w)$, the grounding loss \mathcal{L}_{grnd} is computed as a combination of the L1 loss and the generalized Intersection over Union (gIoU) loss [84], applied on the predicted temporal interval (\hat{c}, \hat{w}) :

$$\begin{aligned} \mathcal{L}_{grnd} = & \lambda_{L1}(|(\hat{c} - c) + |\hat{w} - w|)| \\ & + \lambda_{gIoU} gIoU((\hat{c}, \hat{w}), (c, w)) \end{aligned} \quad (5)$$

3.2.1. MIL framework

A caveat with the pseudo-labeled enriched queries is that they can be noisy and include hallucinations; as a result, some of them may lead to a temporal interval prediction inferior to the original query. To address this, we propose considering *multiple* options for the target query (2 in the case of a single query scenario, Q and Q^{enr}), by adopting a multiple instance learning (MIL) framework [14]. This approach allows the model to choose between the enriched and the original input queries during training, depending on which version leads to a better temporal interval prediction. During inference, this means that our ED-VTG model can choose to either enrich the query if it misses important details, or decide to “carry over” the original query into \hat{Q}^{enr} when it is concrete enough (in which case $Q = \hat{Q}^{enr}$).

Formally, to perform MIL, we collect two temporal interval predictions by running two forward passes with different LLM inputs (in a teacher-forcing fashion), *i.e.* \mathbf{y}^{dir}

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	R@0.3	R@0.5	R@0.7	mIoU
UniVTG [47]	✗	4.2M	ZS	44.1	25.2	10.0	27.1	—	—	—	—	5.2	1.3	0.3	4.4	—	—	—	—
SeViLA [111]	✗	129M	ZS	—	—	—	—	31.6	19.0	10.1	23.0	—	—	—	—	—	—	—	—
PSVL [66]	✗	—	ZS	46.2	31.3	14.2	31.2	44.7	30.1	14.7	29.6	—	—	—	—	—	—	—	—
LT-ZVG [34]	✗	—	ZS	52.9	37.2	19.3	36.0	47.6	32.6	15.4	31.8	—	—	—	—	—	—	—	—
Video-LLaMA° [120]	✓	2.7M	ZS	25.2	10.6	3.4	16.8	21.9	10.8	4.9	16.5	5.1	1.2	0.8	3.4	—	—	—	—
Video-ChatGPT° [58]	✓	100K	ZS	27.2	6.2	1.9	19.7	19.5	10.6	4.8	14.2	6.3	1.7	1.0	4.3	—	—	—	—
Valley [56]	✓	100K	ZS	28.4	1.8	0.3	21.4	30.6	13.7	8.1	21.9	—	—	—	—	—	—	—	—
VideoChat2 [42]	✓	2M	ZS	38.0	14.3	3.8	24.6	40.8	27.8	9.3	27.9	—	—	—	—	—	—	—	—
Momenter [73]	✓	10M	ZS	42.6	26.6	11.6	28.5	42.9	23.0	12.4	29.3	—	—	—	—	—	—	—	—
VTimeLLM° [26]	✓	170K	ZS	51.0	27.5	11.4	31.2	44.0	27.8	<u>14.3</u>	30.4	7.0	1.8	0.8	4.5	—	—	—	—
TimeChat° [82]	✓	125K	ZS	—	32.2	13.4	—	—	—	—	—	6.8	2.1	0.8	4.7	—	—	—	—
HawkEye [102]	✓	715K	ZS	50.6	31.4	14.5	33.7	<u>49.1</u>	<u>29.3</u>	10.7	<u>32.7</u>	—	—	—	—	—	—	—	—
ChatVTG [76]	✓	100K	ZS	<u>52.7</u>	<u>33.0</u>	<u>15.9</u>	<u>34.9</u>	40.7	22.5	9.4	27.2	<u>8.1</u>	<u>3.7</u>	<u>1.3</u>	<u>5.5</u>	—	—	—	—
ED-VTG	✓	136K	ZS	59.5	39.3	19.8	40.2	52.1	33.1	16.0	35.2	14.5	6.0	2.3	12.7	—	—	—	—
△Ours - HawkEye	—	—	ZS	8.9↑	7.9↑	5.3↑	6.5↑	3.0↑	3.8↑	5.3↑	2.5↑	—	—	—	—	—	—	—	—
△Ours - ChatVTG	—	—	ZS	—	—	—	—	—	—	—	—	6.4↑	2.3↑	1.0↑	7.2↑	—	—	—	—

Table 2. **Zero-shot STG results on Charades, ActivityNet, and TACoS test splits.** For all three datasets, ED-VTG gains significant improvement over *all* existing methods, including task-specific, non-generalist models. We use **boldface** for the best and underline the second-best result for each metric, among the generalist models. °Official checkpoints are used for TACoS evaluation.

that is formed using the original query \mathcal{Q} , and \mathbf{y}^{enr} using the pseudo-labeled enriched query \mathcal{Q}^{enr} . We illustrate this schematically in Figure 2 (right). Recall that through the contextualized interval representation \mathbf{h}_{int} , the predicted interval depends on the LLM teacher-forced input. Hence, these inputs produce respective interval predictions, $\hat{\mathbf{I}}^{\text{dir}}$ and $\hat{\mathbf{I}}^{\text{enr}}$. We select the query version that results in the smallest grounding loss and use it to compute our overall training objective:

$$\mathcal{L} = \begin{cases} \lambda_{\text{LM}} \mathcal{L}_{\text{LM}}^{\text{dir}} + \lambda_{\text{grnd}} \mathcal{L}_{\text{grnd}}^{\text{dir}} & \text{if } \mathcal{L}_{\text{grnd}}^{\text{dir}} < \mathcal{L}_{\text{grnd}}^{\text{enr}} \\ \lambda_{\text{LM}} \mathcal{L}_{\text{LM}}^{\text{enr}} + \lambda_{\text{grnd}} \mathcal{L}_{\text{grnd}}^{\text{enr}} & \text{otherwise} \end{cases} \quad (6)$$

where $\mathcal{L}_{\text{LM}}^{\text{dir}}$ and $\mathcal{L}_{\text{LM}}^{\text{enr}}$ are the language losses for targets \mathbf{y}^{dir} and \mathbf{y}^{enr} respectively, and $\mathcal{L}_{\text{grnd}}^{\text{dir}}$ and $\mathcal{L}_{\text{grnd}}^{\text{enr}}$ the grounding losses for predictions $\hat{\mathbf{I}}^{\text{dir}}$ and $\hat{\mathbf{I}}^{\text{enr}}$ respectively. The hyper-parameters λ_{LM} and λ_{grnd} are the relative weights of the language modeling and grounding losses.

4. Experiments

We design experiments to study three key questions related to our architecture and training framework: **Q1**) How does ED-VTG perform on various video grounding tasks in comparison to the current state-of-the-art? **Q2**) How beneficial is our query enrichment approach, within the MIL paradigm, as opposed to directly grounding the original queries? **Q3**) Does utilizing an interval decoder with specifically tailored grounding objectives offer advantages over predicting timestamps as raw text or special tokens?

4.1. Datasets

Table 1 summarizes the datasets that we used during the pre-training and fine-tuning stages. During pre-training, we use a total of 136K medium-to-long duration videos from 8 public datasets annotated with text queries and the corresponding intervals. As a preprocessing step, we collect

pseudo-labels for enriched queries using an external captioning model [39]. In short, we take each video segment defined by the ground-truth intervals and prompt the captioning model to enrich the original query while preserving its meaning given the video segment. We provide full details of this process with the exact prompt used in the supplementary material.

4.2. Tasks

Table 1 also summarizes the tasks for which we use each dataset. Here we briefly describe these tasks, and highlight how our approach is applied to solve them.

Single-Query Temporal Grounding (STG) involves identifying a single time window in response to a single input language query ($N=1$).

Video Paragraph Grounding (VPG) involves grounding $N > 1$ sentences to N corresponding time windows. Our ED-VTG model can be trained to predict multiple enriched queries (one per original input query) interleaved with $\langle \text{INT} \rangle$ tokens that get separately decoded into intervals. Since there are multiple queries in the input, we run multiple forward passes through the LLM to perform MIL, wherein each pass selects a random number of queries to be enriched.

Question Grounding (QG) involves retrieving *evidence intervals* to answer questions, facilitating explainable video QA. As in STG, the input is a single query, and the output a single interval.

Article grounding (AG) is an extension of VPG where the model is given multiple queries as input, some of which may not be groundable in the video. Hence, the model must 1) identify which queries are groundable and 2) predict intervals for the groundable queries.

Note that, while we train our model jointly on multiple tasks, they are unified as a single task in the form of

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) [119]	X	—	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		
MS-2D-TAN (I3D) [124]	X	—	FT	—	56.6	36.2	—	62.1	45.5	28.3	—	42.0	33.6	22.1	—		
Moment-DETR [38]	X	236K	FT	65.8	52.1	30.6	45.5	—	—	—	—	38.0	24.7	12.0	25.5		
UnLoc-B [108]	X	650K	FT	—	58.1	35.4	—	—	48.0	29.7	—	—	—	—	—		
MomentDiff [44]	X	—	FT	—	55.6	32.4	—	—	—	—	—	46.6	28.9	12.4	30.4		
LGI [64]	X	—	FT	73.0	59.5	35.5	51.4	58.5	41.5	23.1	41.1	—	—	—	—		
BAM-DETR [37]	X	—	FT	72.9	60.0	39.4	52.3	—	—	—	—	56.7	41.5	26.8	39.3		
InternVideo2* + CG-DETR [101]	X	2.1M	FT	79.7	70.0	48.9	58.8	—	—	—	—	—	—	—	—		
SG-DETR [23]	X	—	FT	—	71.1	52.8	60.7	—	—	—	—	—	46.4	33.9	42.4		
EMB (ELA) [29]	X	—	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) [40]	✓	129M	FT	—	43.3	<u>32.6</u>	—	—	25.8	9.7	—	—	—	—	—		
VideoChat2 [42]	✓	2M	FT	—	—	—	—	<u>55.5</u>	<u>34.7</u>	17.7	38.9	—	—	—	—		
TimeChat [82]	✓	125K	FT	—	46.7	23.7	—	—	—	—	—	<u>27.7</u>	<u>15.1</u>	<u>6.4</u>	<u>18.4</u>		
HawkEye [102]	✓	715K	FT	<u>72.5</u>	<u>58.3</u>	28.8	<u>49.3</u>	<u>55.9</u>	<u>34.7</u>	<u>17.9</u>	<u>39.1</u>	—	—	—	—		
VtimeLLM [26]	✓	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	<u>5.7 ↑</u>	<u>3.8 ↑</u>	<u>6.2 ↑</u>	<u>3.3 ↑</u>	<u>11.7 ↑</u>	<u>10.4 ↑</u>	<u>4.8 ↑</u>	<u>5.8 ↑</u>	—	—	—	—		
$\Delta_{\text{Ours - VTimeLLM}}$	—	—	FT	—	—	—	—	—	—	—	—	<u>19.2 ↑</u>	<u>17.1 ↑</u>	<u>9.7 ↑</u>	<u>14.4 ↑</u>		

Table 3. **Fine-tuned STG results on Charades, ActivityNet, and TACoS test splits.** For all datasets, ED-VTG achieves strong improvements over previous generalist models, and performs comparably to task-specific expert models. *Though InterVideo2 is a generalist model, it fine-tunes CG-DETR [62] head for grounding tasks, using the LLM as a feature extractor. For completeness, we report all competitive existing works, including task-specific SOTA specialist models (shown in gray), but directly compare ED-VTG only with generalist frameworks - using **boldface** for the best and underlining the second best results among generalist models (see discussion in Section 4.3).

(Video, Text) → (Time intervals). Besides the differences in training data sources, we use the identical training objective (Eq. 6) during pre-training and fine-tuning stages.

4.3. Evaluation Protocol

Following the common practices in the literature [26, 82, 102], we evaluate ED-VTG in two primary evaluation protocols: (i) **Zero-shot (ZS)**, where the pre-trained model is assessed directly without any fine-tuning on downstream datasets, and (ii) **Fine-tuned (FT)**, where the model undergoes additional training on specific tasks and datasets. We also present results without pre-training to assess its importance. For the STG and VPG tasks, following existing works [7, 26, 82, 87, 102], we report the mean intersection over union (mIoU) and Recall@1 for IoU $\geq m$ (R@m), with $m \in \{0.3, 0.5, 0.7\}$. For the QG task, we report intersection over prediction (IoP) in addition to IoU, following the NeXT-GQA [107] evaluation protocol. For AG, we adhere to the HT-Step [3] protocol and report article-grounding mAP scores across various IoU thresholds.

Comparison to non-generalist models. In the results section, we report all existing SOTA and competitive methods for a complete comparison, however we divide them into generalist and non-generalist (specialist) models. We note that specialist methods are heavily tailored to the task and in practice often overfit specific datasets which limits their transferability (as is evidenced by the zero-shot results in Table 2). We therefore focus the discussion around our method’s comparison to other generalist models.

4.4. Implementation Details

We initialize the video encoder and LLM with the Video-LLAMA-7B [120] checkpoint, which is a similarly sized backbone used in existing LLM-based video grounding

models [26, 73, 82, 102]. Video-LLAMA is trained on video captioning tasks with WebVid [6] and VideoChat [41]. The video encoder includes a ViT-G/14 from EVA-CLIP [93] as the frame feature extractor, followed by image and video QFormers. We initialize the decoder with random weights. We keep the ViT frozen, apply LoRA [25] with rank 32 to the LLM and fully tune the Q-Formers, decoder, and linear layers. We use LLAVA OneVision (OV) 72B [39] as the external captioner for obtaining enriched queries in the training sets. For the VPG task, we run four forward passes to perform MIL. We pre-train our model for 40 epochs with a batch size 256, using AdamW [53] with a peak learning rate of 5e-5 and a cosine scheduler [52] with a linear warmup for the first 20% steps. Pre-training takes 2 days on 16 V100 nodes (8 cards with 32G GPU memory each). Additional details on pre-training, fine-tuning, and task-specific instructions are provided in the supplementary material.

4.5. Results

Single-Query Temporal Grounding (STG). We start by comparing ED-VTG against the state-of-the-art across three different STG benchmarks, namely Charades-STA, ActivityNet-Captions, and TACoS, in a zero-shot evaluation setting. We show the results in Table 2. On Charades, ED-VTG achieves ZS scores of 59.5, 39.3, and 19.8 for R@0.3, R@0.5, and R@0.7, respectively, significantly outperforming all baseline models. Notably, ED-VTG surpasses Momenter [73] and HawkEye [102] by **11.4** and **6.2** absolute mIoU points despite these models being pre-trained with 100x and 6x more segment-level data, respectively. A similar trend is observed on ActivityNet and TACoS where our model achieves improvements of 2.5 and 7.2 mIoU points over the nearest LLM-based models in zero-shot setting. Notably, the TACoS dataset con-

Method	Generalist Model	Charades-CD-OOD				Method	Generalist Model	ANet-Captions				Method	Generalist Model	TACoS				Method	Generalist Model	YouCook2			
		R@0.3	R@0.5	mIoU				R@0.3	R@0.5	mIoU			R@0.3	R@0.5	mIoU			R@0.3	R@0.5	mIoU			
DepNet [7]	X	45.6	27.6	29.3	CBLN [48]	X	66.3	48.1	27.6	CMIN [125]	X	24.6	18.1	—	DORi [86]	X	43.4	30.5	30.5				
DRN [115]	X	40.5	30.4	—	2D-TAN [122]	X	59.5	44.5	—	2D-TAN [122]	X	37.3	25.3	—	DORi* [86]	X	42.3	29.9	29.9				
STLG [55]	X	48.3	30.4	—	3D-TPN [123]	X	67.6	51.5	—	LocFormer [87]	X	46.8	31.3	30.9	LocFormer [87]	X	46.8	31.3	30.9				
SVPTR [32]	X	50.3	28.5	32.1	DepNet [7]	X	72.8	55.9	—	ExCL [21]	X	26.6	16.2	18.9	DepNet [7]	X	26.6	16.2	18.9				
SiamGTR [94]	X	59.1	35.5	38.9	SVPTR [32]	X	78.1	61.7	55.9	TMLGA [85]	X	47.9	28.2	31.4	TMLGA [85]	X	34.8	23.1	24.4				
VTimeLLM [†] [26]	✓	53.2	34.0	35.1	VTimeLLM [†] [26]	✓	66.1	50.3	45.6	VTimeLLM [†] [26]	✓	40.2	25.6	27.9	VTimeLLM [†] [26]	✓	41.3	18.5	24.3				
TimeChat [†] [82]	✓	60.5	36.1	38.3	TimeChat [†] [82]	✓	67.9	51.5	47.0	TimeChat [†] [82]	✓	39.5	25.6	27.8	TimeChat [†] [82]	✓	40.9	19.0	26.6				
ED-VTG	✓	70.7	47.3	45.0	ED-VTG	✓	74.1	58.0	53.7	ED-VTG	✓	46.2	27.8	30.7	ED-VTG	✓	48.1	28.0	31.5				
$\Delta_{\text{Ours} - \text{TimeChat}}$	—	10.2↑	11.2↑	6.7↑	$\Delta_{\text{Ours} - \text{TimeChat}}$	—	6.2↑	6.5↑	6.7↑	$\Delta_{\text{Ours} - \text{TimeChat}}$	—	6.7↑	2.2↑	2.9↑	$\Delta_{\text{Ours} - \text{TimeChat}}$	—	7.2↑	9.0↑	4.9↑				

(a) Results on Charades-CD-OOD.

(b) Results on ActivityNet.

(c) Results on TACoS.

(d) Results on YouCook2.

Table 4. Performance on VPG task on four different benchmarks: Charades-CD-OOD, ActivityNet-Captions, TACoS, YouCook2. ED-VTG significantly improves over previous LLM-based models, and performs comparably to state-of-the-art specialist models. [†]We fine-tune VTimeLLM and TimeChat checkpoints for the VPG task. Dori* represents frozen text (BERT) encoder during fine-tuning.

Method	Generalist Model	NeXT-GQA					
		mIoP	IoP@0.3	IoP@0.5	mIoU	IoU@0.3	IoU@0.5
VGT [106]	X	24.7	26.0	24.6	3.0	4.2	1.4
VIOLETv2 [17]	X	23.6	25.1	23.3	3.1	4.3	1.3
Temp(CLIP) NG+ [107]	X	25.7	31.4	25.5	12.1	17.5	8.9
FrozenBiLM NG+ [109]	X	24.2	28.5	23.7	9.6	13.5	6.1
SeViLA [111]	X	29.5	34.7	22.9	21.7	29.2	13.8
LLoVi 7B [°] [117]	✓	20.7	—	20.5	8.7	—	6.0
VideoStreaming [°] [74]	✓	32.2	—	31.0	19.3	—	13.3
LongRepo 7B [°] [33]	✓	20.3	—	20.0	8.7	—	6.0
DeVi [75]	✓	33.8	—	32.2	20.7	17.4	—
HawkEye [102]	✓	—	—	—	25.7	37.0	19.5
ED-VTG	✓	34.7	45.1	33.5	26.6	39.5	19.8
$\Delta_{\text{Ours} - \text{SeViLA}}$	—	5.2↑	10.4↑	10.6↑	4.9↑	10.3↑	6.0↑
$\Delta_{\text{Ours} - \text{HawkEye}}$	—	—	—	—	0.9↑	2.5↑	0.3↑

Table 5. Performance on QG task on NeXT-GQA test split. ED-VTG consistently achieves consistent improvements over the existing models across all metrics. [°]Results of LLoVi, LongRepo and VideoStreaming are from [74].

Method	Seen				Unseen			
	↑ mAP@IoU @0.3	↑ mAP@IoU @0.5	↑ mAP@IoU @0.7	↑ mAP@IoU @[0.3-0.7]	↑ mAP@IoU @0.3	↑ mAP@IoU @0.5	↑ mAP@IoU @0.7	↑ mAP@IoU @[0.3-0.7]
UMT [51]	15.7	8.7	3.2	9.1	9.4	4.9	1.7	5.3
MT+BCE [3, 59]	46.2	29.9	12.9	29.8	31.6	18.7	7.7	19.3
ActionFormer-T [118]	41.2	30.8	18.3	30.2	29.7	20.3	10.7	20.4
TimeChat [†] [82]	45.3	29.0	14.4	29.0	30.7	17.8	7.5	18.7
ED-VTG	48.9	31.5	18.0	32.5	33.0	21.2	11.1	21.6
$\Delta_{\text{Ours} - \text{TimeChat}}$	3.6↑	2.5↑	3.6↑	3.5↑	2.3↑	3.4↑	3.6↑	2.9↑

Table 6. Performance on AG task on HT-Step seen and unseen val split. ED-VTG is the first LLM-based model to report results on for video grounding in the presence of negative, non-groundable queries and sets a new state of the art in terms of average mAP score across various IoU thresholds. [†]We fine-tune the official TimeChat checkpoint for AG task.

tains short, under-specified queries, longer input videos, and fine-grained interval annotations, which pose challenges for LLMs with a fixed number of input frames. The enriched query descriptions that ED-VTG generates enable it to more accurately align them to video frames and precisely retrieve the correct intervals.

On the fine-tuned STG evaluation, as shown in Table 3, ED-VTG demonstrates an impressive gain of 5.7 and 11.7 points in R@0.3 over HawkEye on Charades and ActivityNet, respectively. Similarly on TACoS, ED-VTG surpasses all existing MLLMs by a considerable margin, e.g. 14.0 and 14.4 mIoU points over TimeChat and VTimeLLM. Furthermore, ED-VTG also beats many existing task-specific specialist models in all three benchmarks, significantly reducing the gap between specialist models

and MLLMs for STG.

Video Paragraph Grounding (VPG). Next, we fine-tune ED-VTG for the VPG task, where the model processes multiple input queries in a temporal sequence. While specialist models have reported results in the past, no LLM-based models have previously addressed this challenging task. To establish a baseline for comparison, we fine-tune the officially released VTimeLLM and TimeChat pre-trained checkpoints. As shown in Table 4a, ED-VTG significantly outperforms both LLMs on the Charades-CD-OOD dataset, achieving an absolute gain of 9.9 and 6.7 mIoU points over VTimeLLM and TimeChat, respectively. Additionally, ED-VTG surpasses all specialist models on this dataset by a substantial margin, setting a new state-of-the-art.

We also evaluate VPG performance on three other benchmarks: ActivityNet-Captions, TACoS, and YouCook2, as presented in Tables 4b, 4c, and 4d. Consistent with the results on Charades, ED-VTG outperforms LLM-based models across all metrics on these datasets, with mIoU gains of 6.7, 2.9, and 4.9 over TimeChat on ActivityNet-Captions, TACoS, and YouCook2, respectively. ED-VTG also exceeds many existing specialist VPG models, demonstrating the effectiveness of our enriched queries and cascaded interval decoder.

Question Grounding (QG). To further assess the model’s generalization capabilities, we conduct zero-shot evaluation on the held-out QG task. Our results on the NeXT-GQA test set are presented in Table 5. Notably, ED-VTG achieves state-of-the-art performance across all metrics, outperforming both specialist and LLM-based models by a significant margin. Specifically, ED-VTG surpasses the existing best baseline, HawkEye, by 2.5 points in terms of IoU@0.3 score, demonstrating its strong generalizability.

Article Grounding (AG). We assess ED-VTG on the AG on the HT-Step benchmark. Table 6 shows the fine-tuned performance on the AG task, where ED-VTG shows significant improvements over existing baselines. Notably, on the challenging *unseen* split, our model achieves the best results across all metrics, surpassing both the LLM and specialist models by a decent margin. The ability to handle non-groundable queries underscores ED-VTG’s real-world

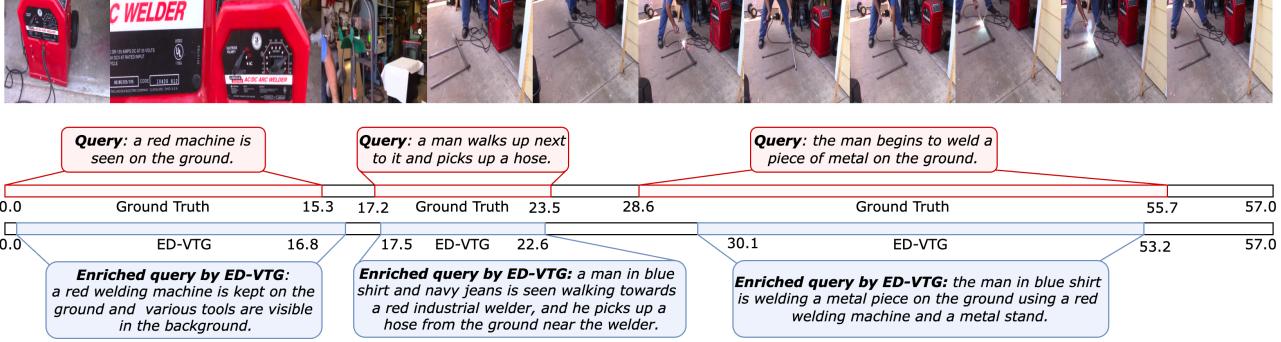


Figure 3. Example of query enrichment and detection made by ED-VTG on video paragraph grounding (VPG) task from the ActivityNet-Captions [36] dataset. In this specific sample, three different queries are enriched and localized together.

Training Paradigm	Charades-STA STG												ActivityNet-Captions STG																								
	ZS			FT w/o PT			FT			ZS			FT w/o PT			FT			R@0.3			R@0.5			mIoU												
	R@0.3	R@0.5	mIoU	R@0.3	R@0.5	mIoU	R@0.3	R@0.5	mIoU	R@0.3	R@0.5	mIoU	R@0.3	R@0.5	mIoU	R@0.3	R@0.5	mIoU	R@0.3	R@0.5	mIoU	R@0.3	R@0.5	mIoU													
Detect	48.1	30.6	31.0	51.4	31.5	33.2	68.9	49.0	45.8	46.3	26.0	29.6	50.3	30.1	34.0	61.1	38.0	39.2	58.1	37.3	37.7	60.1	37.0	38.4	75.1	56.6	49.7	50.7	29.5	33.4	56.3	35.5	37.8	65.5	43.4	43.8	
Enrich & Detect	59.5	39.3	40.2	62.8	38.4	40.3	78.2	62.1	52.6	52.1	33.1	35.2	57.5	36.2	38.6	67.6	45.1	44.9	59.5	39.3	40.2	62.8	38.4	40.3	78.2	62.1	52.6	52.1	33.1	35.2	57.5	36.2	38.6	67.6	45.1	44.9	
Enrich & Detect w/ MIL																																					

Table 7. Ablation on the effect of enriched queries. Our proposed enrich & detect framework significantly gains over directly grounding the input queries across different evaluation settings. Introducing the MIL framework further improve the performance. FT w/o PT refers directly fine-tuning on respective datasets, without performing pre-training.

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 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 8. Ablation on enrichment as a training pre-processing step. The two step enrich & detect framework is more helpful since the trained model learn to perform autonomous enrichment during evaluation. Reported results are in FT w/o PT setting.

Decoder	Objectives			Charades-STA STG			ANet-Captions STG		
	LM	L1	gIoU	R@0.3	R@0.5	mIoU	R@0.3	R@0.5	mIoU
—	✓	—	—	54.2	33.2	34.1	51.0	31.6	35.5
✓	✓	✓	—	58.5	36.0	37.0	55.8	34.4	37.1
✓	✓	—	✓	58.9	36.2	37.1	55.8	34.7	37.3
✓	✓	✓	✓	60.1	37.0	38.4	56.3	35.5	37.8

Table 9. Ablation study on different training objectives. Training the model only using the LM loss (without the decoder) leads to significant performance drop. Results are in FT w/o PT setting.

applicability, as it does not always assume that the query is occurring in the input video.

4.6. Ablation Study

Effect of Enriched Queries. We examine the impact of query enrichment through a step-by-step ablation, as shown in Tables 7 and 8. Initially, we compare the effect of enrichment without MIL paradigm against direct grounding. As indicated in the first two rows of Table 7, enriched queries lead to significant improvements on the Charades-STA and ActivityNet-Captions STG benchmarks. In the zero-shot (ZS) setting, enrichment results in gains of 6.7 and 3.8 mIoU points on these datasets, respectively. Introducing the

MIL paradigm further enhances performance, adding 2.5 and 1.8 mIoU points. Similar improvements are observed in other evaluation settings, highlighting the substantial effectiveness of query enrichment within the MIL framework.

Additionally, we investigate the effect of offline enrichment in Table 8. In this scenario, instead of using a two-step grounding process, we enrich the queries as a training enrichment step, and the model is then directly provided with these enriched queries as input and asked to directly perform grounding. However, we find that offline enrichment is not advantageous, primarily due to the lack of enrichment during evaluation. In contrast, our two-step grounding approach allows the trained model to learn how to enrich queries and improve them autonomously when necessary, resulting in significant performance gains.

Training Objectives. We ablate different training objectives on the Charades and ActivityNet STG benchmarks, as shown in Table 9. The decoder achieves optimal performance when both L1 and gIoU objectives are used together; omitting either one slightly reduces the scores.

4.7. Qualitative Results and Error Analysis

Figure 3 visualizes a VPG sample from the ActivityNet-Captions dataset where ED-VTG meaningfully enriches the three input queries, then proceeds to precisely ground them. Please refer to supplementary for more qualitative results, including intuitive demonstrations of the flexibility in choosing between enrichment or direct detection, and comparison with TimeChat baseline. We also present there some interesting failure cases, such as the ones involving small and obscured objects in long input videos.

5. Conclusion

In this paper, we presented ED-VTG, a novel method for fine-grained video temporal grounding using multi-modal LLMs. By enhancing queries with additional details, utilizing a lightweight decoder and trained in a multiple-instance framework, ED-VTG accurately locates temporal boundaries in videos. Our experiments show that our method outperforms existing LLM-based works and is competitive with specialized models, especially in zero-shot settings, setting a new, strong benchmark for video grounding tasks.

6. Acknowledgement

This codebase for this project is built on the TimeChat [82] and VTimeLLM [26] repository. We would like to thank the respective authors for their contributions. We gratefully acknowledge the following colleagues at FAIR for valuable discussions and support of our project: Tushar Nagarajan, Huiyu Wang, Yujie Lu, and Arjun Somayazulu.

References

- [1] Moloud Abdar, Meenakshi Kollati, Swaraja Kuraparthi, Farhad Pourpanah, Daniel McDuff, Mohammad Ghavamzadeh, Shuicheng Yan, Abdullah Mohamed, Abbas Khosravi, Erik Cambria, et al. A review of deep learning for video captioning. *arXiv preprint arXiv:2304.11431*, 2023. [1](#), [4](#)
- [2] 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. [17](#)
- [3] Triantafyllos Afouras, Effrosyni Mavroudi, Tushar Nagarajan, Huiyu Wang, and Lorenzo Torresani. Ht-step: Aligning instructional articles with how-to videos. In *NeurIPS*, 2024. [4](#), [6](#), [7](#), [21](#), [22](#)
- [4] Stuart Andrews, Ioannis Tsochantaridis, and Thomas Hofmann. Support vector machines for multiple-instance learning. In *NeurIPS*, pages 561–568, 2003. [3](#)
- [5] 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. [4](#), [18](#)
- [6] Max Bain, Arsha Nagrani, GüL Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *ICCV*, pages 1728–1738, 2021. [6](#)
- [7] Peijun Bao, Qian Zheng, and Yadong Mu. Dense events grounding in video. In *AAAI*, pages 920–928, 2021. [2](#), [6](#), [7](#), [21](#)
- [8] Hakan Bilen and Andrea Vedaldi. Weakly supervised deep detection networks. In *CVPR*, pages 2846–2854, 2016. [3](#)
- [9] 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. [17](#), [21](#)
- [10] 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. [17](#)
- [11] Yi-Wen Chen, Yi-Hsuan Tsai, and Ming-Hsuan Yang. End-to-end multi-modal video temporal grounding. *NeurIPS*, 34:28442–28453, 2021. [1](#)
- [12] Ramazan Cinbis, Jakob Verbeek, and Cordelia Schmid. Weakly supervised object localization with multi-fold multiple instance learning. *IEEE TPAMI*, 39, 2015. [3](#)
- [13] Chaorui Deng, Shizhe Chen, Da Chen, Yuan He, and Qi Wu. Sketch, ground, and refine: Top-down dense video captioning. In *CVPR*, pages 234–243, 2021. [2](#)
- [14] Thomas G. Dietterich, Richard H. Lathrop, and Tomás Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1):31–71, 1997. [3](#), [4](#)
- [15] Li Dong, Jonathan Mallinson, Siva Reddy, and Mirella Lapata. Learning to paraphrase for question answering. In *EMNLP*, 2017. [2](#)
- [16] 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. [14](#), [17](#), [20](#)
- [17] Tsu-Jui Fu, Linjie Li, Zhe Gan, Kevin Lin, William Yang Wang, Lijuan Wang, and Zicheng Liu. An empirical study of end-to-end video-language transformers with masked visual modeling. In *CVPR*, pages 22898–22909, 2023. [2](#), [7](#)
- [18] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. Tall: Temporal activity localization via language query. In *ICCV*, pages 5267–5275, 2017. [4](#), [15](#), [16](#), [19](#), [20](#), [21](#)
- [19] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. Tall: Temporal activity localization via language query. In *ICCV*, pages 5267–5275, 2017. [17](#)
- [20] Silin Gao, Yichi Zhang, Zhijian Ou, and Zhou Yu. Paraphrase augmented task-oriented dialog generation. In *ACL*, pages 639–649, 2020. [3](#)
- [21] Soham Ghosh, Anuva Agarwal, Zarana Parekh, and Alexander G Hauptmann. Excl: Extractive clip localization using natural language descriptions. In *NAACL*, pages 1984–1990, 2019. [2](#), [7](#)
- [22] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Region-based convolutional networks for accurate object detection and segmentation. *IEEE TPAMI*, 38(1):142–158, 2015. [2](#)
- [23] Aleksandr Gordeev, Vladimir Dokholyan, Irina Tolstykh, and Maksim Kuprashevich. Saliency-guided detr for moment retrieval and highlight detection. *arXiv preprint arXiv:2410.01615*, 2024. [6](#), [16](#), [17](#)
- [24] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, pages 2961–2969, 2017. [2](#)
- [25] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *ICLR*, 2022. [6](#)

- [26] Bin Huang, Xin Wang, Hong Chen, Zihan Song, and Wenwu Zhu. Vtimellm: Empower llm to grasp video moments. In *CVPR*, pages 14271–14280, 2024. 2, 5, 6, 7, 9, 16, 17, 21
- [27] 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. 2, 16
- [28] Gabriel Huang, Bo Pang, Zhenhai Zhu, Clara Rivera, and Radu Soricut. Multimodal pretraining for dense video captioning. In *AACL*, pages 470–490, 2020. 4, 20
- [29] Jiabo Huang, Hailin Jin, Shaogang Gong, and Yang Liu. Video activity localisation with uncertainties in temporal boundary. In *ECCV*, pages 724–740. Springer, 2022. 6, 16, 17
- [30] Vladimir Iashin and Esa Rahtu. Multi-modal dense video captioning. In *CVPR Workshops*, 2020. 2
- [31] 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. 16, 17
- [32] 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. 2, 7, 21
- [33] Kumara Kahatapitiya, Kanchana Ranasinghe, Jongwoo Park, and Michael S Ryoo. Language repository for long video understanding. *arXiv preprint arXiv:2403.14622*, 2024. 7
- [34] Dahye Kim, Jungin Park, Jiyoung Lee, Seongheon Park, and Kwanghoon Sohn. Language-free training for zero-shot video grounding. In *WACV*, pages 2539–2548, 2023. 5
- [35] Mahnaz Koupaei and William Yang Wang. Wikihow: A large scale text summarization dataset. *arXiv preprint arXiv:1810.09305*, 2018. 22
- [36] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *ICCV*, pages 706–715, 2017. 2, 4, 8, 16, 21
- [37] 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. 6, 17
- [38] Jie Lei, Tamara L Berg, and Mohit Bansal. Detecting moments and highlights in videos via natural language queries. *NeurIPS*, 34:11846–11858, 2021. 1, 2, 6, 17
- [39] 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. 3, 4, 5, 6, 14, 16, 17, 20
- [40] 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. 6, 17
- [41] KunChang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. Videochat: Chat-centric video understanding. *arXiv preprint arXiv:2305.06355*, 2023. 6
- [42] 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. 2, 5, 6, 17
- [43] Lei Li, Yuwei Yin, Shicheng Li, Liang Chen, Peiyi Wang, Shuhuai Ren, Mukai Li, Yazheng Yang, Jingjing Xu, Xu Sun, et al. M³IT: A Large-Scale Dataset towards Multi-Modal Multilingual Instruction Tuning. *arXiv preprint arXiv:2306.04387*, 2023. 17
- [44] 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. 6, 16, 17, 20
- [45] Yehao Li, Ting Yao, Yingwei Pan, Hongyang Chao, and Tao Mei. Jointly localizing and describing events for dense video captioning. *CVPR*, pages 7492–7500, 2018. 2
- [46] 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. 2, 16
- [47] Kevin Qinghong Lin, Pengchuan Zhang, Joya Chen, Shravan 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. 1, 2, 5, 16, 17, 20
- [48] 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. 2, 7, 21
- [49] 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. 17
- [50] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *ECCV*, pages 21–37, 2016. 4
- [51] 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. 7, 17
- [52] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. In *ICLR*, 2017. 6, 20
- [53] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 6, 20
- [54] Jérôme Louradour. whisper-timestamped., 2023. 20
- [55] Fan Luo, Shaoxiang Chen, Jingjing Chen, Zuxuan Wu, and Yu-Gang Jiang. Self-supervised learning for semi-supervised temporal language grounding. *IEEE Transactions on Multimedia*, 25:7747–7757, 2022. 2, 7
- [56] Ruipu Luo, Ziwang Zhao, Min Yang, Junwei Dong, Da Li, Pengcheng Lu, Tao Wang, Linmei Hu, Minghui Qiu, and Zhongyu Wei. Valley: Video assistant with large language model enhanced ability. *arXiv preprint arXiv:2306.07207*, 2023. 5

- [57] Kaijing Ma, Xianghao Zang, Zerun Feng, Han Fang, Chao Ban, Yuhang 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. [2](#), [16](#)
- [58] Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. Video-chatgpt: Towards detailed video understanding via large vision and language models. In *ACL*, 2024. [5](#)
- [59] Effrosyni Mavroudi, Triantafyllos Afouras, and Lorenzo Torresani. Learning to ground instructional articles in videos through narrations. In *ICCV*, pages 15201–15213, 2023. [7](#)
- [60] Sachit Menon and Carl Vondrick. Visual classification via description from large language models. *ICLR*, 2023. [3](#)
- [61] 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. [22](#)
- [62] 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. [6](#), [16](#), [17](#)
- [63] 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. [1](#), [2](#), [16](#), [17](#), [20](#)
- [64] Jonghwan Mun, Minsu Cho, and Bohyung Han. Local-global video-text interactions for temporal grounding. In *CVPR*, pages 10810–10819, 2020. [6](#), [17](#)
- [65] 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. [4](#), [20](#)
- [66] Jinwoo Nam, Daechul Ahn, Dongyeop Kang, Seong Jong Ha, and Jonghyun Choi. Zero-shot natural language video localization. In *ICCV*, pages 1470–1479, 2021. [5](#)
- [67] Phuc Xuan Nguyen, Tianzhu Liu, Gaurav Prasad, Hung Hai Bui, Binh Pham, and Svetha Venkatesh. Weakly supervised action localization by sparse temporal pooling network. In *CVPR*, pages 6752–6761, 2019. [3](#)
- [68] 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. [4](#), [18](#)
- [69] Boxiao Pan, Haoye Cai, De-An Huang, Kuan-Hui Lee, Adrien Gaidon, Ehsan Adeli, and Juan Carlos Niebles. Spatio-temporal graph for video captioning with knowledge distillation. In *CVPR*, pages 10870–10879, 2020. [1](#)
- [70] Sujoy Paul, Sourya Roy, and Amit K Roy-Chowdhury. W-talc: Weakly-supervised temporal activity localization and classification. In *ECCV*, pages 563–579, 2018. [3](#)
- [71] 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. [17](#)
- [72] Archiki Prasad, Elias Stengel-Eskin, and Mohit Bansal. Rephrase, augment, reason: Visual grounding of questions for vision-language models. In *ICLR*, 2024. [2](#)
- [73] Long Qian, Juncheng Li, Yu Wu, Yaobo Ye, Hao Fei, Tat-Seng Chua, Yueteng Zhuang, and Siliang Tang. Momentor: Advancing video large language model with fine-grained temporal reasoning. In *ICML*, 2024. [2](#), [5](#), [6](#), [16](#)
- [74] Rui Qian, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Shuangrui Ding, Dahua Lin, and Jiaqi Wang. Streaming long video understanding with large language models. *NeurIPS*, 37:119336–119360, 2024. [7](#)
- [75] Hangyu Qin, Junbin Xiao, and Angela Yao. Question-answering dense video events. *arXiv preprint arXiv:2409.04388*, 2024. [7](#)
- [76] Mengxue Qu, Xiaodong Chen, Wu Liu, Alicia Li, and Yao Zhao. Chatvtg: Video temporal grounding via chat with video dialogue large language models. In *CVPR*, pages 1847–1856, 2024. [5](#)
- [77] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavy, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *ICML*, pages 28492–28518. PMLR, 2023. [20](#)
- [78] Joseph Redmon and Ali Farhadi. Yolo9000: Better, faster, stronger. In *CVPR*, pages 6517–6525, 2017. [4](#)
- [79] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, pages 779–788, 2016. [4](#)
- [80] Michaela Regneri, Marcus Rohrbach, Dominikus Wetzel, Stefan Thater, Bernt Schiele, and Manfred Pinkal. Grounding action descriptions in videos. *TACL*, 1:25–36, 2013. [4](#), [19](#), [21](#)
- [81] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE TPAMI*, 39(6):1137–1149, 2016. [2](#)
- [82] 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](#), [5](#), [6](#), [7](#), [9](#), [14](#), [15](#), [16](#), [17](#), [20](#), [21](#)
- [83] Video Description Research and Development Center. Youdescribe, 2013. [18](#)
- [84] Seyed Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian D. Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. *CVPR*, pages 658–666, 2019. [4](#)
- [85] Cristian Rodriguez, Edison Marrese-Taylor, Fatemeh Sadat Saleh, Hongdong Li, and Stephen Gould. Proposal-free temporal moment localization of a natural-language query in video using guided attention. In *WACV*, pages 2464–2473, 2020. [2](#), [7](#)
- [86] Cristian Rodriguez-Opazo, Edison Marrese-Taylor, Basura Fernando, Hongdong Li, and Stephen Gould. Dori: Discovering object relationships for moment localization of a natural language query in a video. In *WACV*, pages 1079–1088, 2021. [2](#), [7](#)
- [87] Cristian Rodriguez-Opazo, Edison Marrese-Taylor, Basura Fernando, Hiroya Takamura, and Qi Wu. Memory-efficient

- temporal moment localization in long videos. In *EACL*, pages 1909–1924, 2023. [2](#), [6](#), [7](#)
- [88] 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. [21](#)
- [89] T-YLPG Ross and GKHP Dollár. Focal loss for dense object detection. In *CVPR*, pages 2980–2988, 2017. [2](#)
- [90] Himanshu Sharma, Manmohan Agrahari, Sujeeet Kumar Singh, Mohd Firoj, and Ravi Kumar Mishra. Image captioning: a comprehensive survey. In *2020 International Conference on Power Electronics & IoT Applications in Renewable Energy and its Control (PARC)*, pages 325–328. IEEE, 2020. [1](#), [4](#)
- [91] Matteo Stefanini, Marcella Cornia, Lorenzo Baraldi, Silvia Cascianelli, Giuseppe Fiameni, and Rita Cucchiara. From show to tell: A survey on deep learning-based image captioning. *IEEE TPAMI*, 45(1):539–559, 2022. [1](#), [4](#)
- [92] 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. [1](#), [17](#), [20](#)
- [93] 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. [6](#), [20](#)
- [94] Chaolei Tan, Jianhuang Lai, Wei-Shi Zheng, and Jian-Fang Hu. Siamese learning with joint alignment and regression for weakly-supervised video paragraph grounding. In *CVPR*, pages 13569–13580, 2024. [2](#), [7](#)
- [95] 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. [4](#), [18](#)
- [96] Damien Teney, Ehsan Abbasnejad, and Anton van den Hengel. Unshuffling data for improved generalization in visual question answering. In *ICCV*, pages 1397–1407, 2021. [2](#)
- [97] 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. [18](#)
- [98] 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. [1](#), [2](#), [16](#)
- [99] 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. [17](#)
- [100] 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. [17](#)
- [101] 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. [6](#), [17](#)
- [102] 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. [5](#), [6](#), [7](#), [17](#)
- [103] 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. [21](#)
- [104] Hao Wu, Huabin Liu, Yu Qiao, and Xiao Sun. Dibs: Enhancing dense video captioning with unlabeled videos via pseudo boundary enrichment and online refinement. In *CVPR*, pages 18699–18708, 2024. [2](#)
- [105] 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. [21](#)
- [106] Junbin Xiao, Pan Zhou, Tat-Seng Chua, and Shuicheng Yan. Video graph transformer for video question answering. In *ECCV*, pages 39–58. Springer, 2022. [2](#), [7](#)
- [107] 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. [2](#), [4](#), [6](#), [7](#), [21](#)
- [108] Shen Yan, Xuehan Xiong, Arsha Nagrani, Anurag Arnab, Zhonghao Wang, Weinan Ge, David Ross, and Cordelia Schmid. Unloc: A unified framework for video localization tasks. In *CVPR*, pages 13623–13633, 2023. [6](#), [16](#), [17](#)
- [109] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Zero-shot video question answering via frozen bidirectional language models. In *NeurIPS*, pages 124–141, 2022. [2](#), [7](#)
- [110] Antoine Yang, Arsha Nagrani, Paul Hongseok 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. [2](#), [20](#)
- [111] Shoubin Yu, Jaemin Cho, Prateek Yadav, and Mohit Bansal. Self-chained image-language model for video localization and question answering. *NeurIPS*, 36, 2024. [2](#), [5](#), [7](#)
- [112] 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. [4](#), [21](#)
- [113] 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. [4](#), [18](#)
- [114] 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. [4](#), [20](#)

- [115] Runhao Zeng, Haoming Xu, Wenbing Huang, Peihao Chen, Mingkui Tan, and Chuang Gan. Dense regression network for video grounding. In *CVPR*, pages 10287–10296, 2020. [2](#), [7](#)
- [116] Xiangyu Zeng, Kunchang Li, Chenting Wang, Xinhao Li, Tianxiang Jiang, Ziang Yan, Songze Li, Yansong Shi, Zhengrong Yue, Yi Wang, et al. Timesuite: Improving mllms for long video understanding via grounded tuning. In *ICLR*, 2025. [2](#)
- [117] Ce Zhang, Taixi Lu, Md Mohaiminul Islam, Ziyang Wang, Shoubin Yu, Mohit Bansal, and Gedas Bertasius. A simple llm framework for long-range video question-answering. In *EMNLP*, pages 21715–21737, 2024. [7](#)
- [118] Chen-Lin Zhang, Jianxin Wu, and Yin Li. Actionformer: Localizing moments of actions with transformers. In *ECCV*, pages 492–510. Springer, 2022. [7](#)
- [119] 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. [1](#), [2](#), [6](#), [17](#)
- [120] Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. In *EMNLP*, pages 543–553, 2023. [5](#), [6](#)
- [121] Junchao Zhang and Yuxin Peng. Object-aware aggregation with bidirectional temporal graph for video captioning. In *CVPR*, pages 8327–8336, 2019. [1](#)
- [122] 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. [2](#), [7](#), [17](#)
- [123] 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. [2](#), [7](#)
- [124] 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. [6](#), [17](#)
- [125] Zhu Zhang, Zhijie Lin, Zhou Zhao, and Zhenxin Xiao. Cross-modal interaction networks for query-based moment retrieval in videos. In *ACM SIGIR*, pages 655–664, 2019. [7](#)
- [126] Yue Zhao, Ishan Misra, Philipp Krähenbühl, and Rohit Girdhar. Learning video representations from large language models. In *CVPR*, pages 6586–6597, 2023. [3](#)
- [127] Luowei Zhou, Chenliang Xu, and Jason Corso. Towards automatic learning of procedures from web instructional videos. In *AAAI*, 2018. [4](#), [21](#)
- [128] Luowei Zhou, Chenliang Xu, and Jason J Corso. Towards automatic learning of procedures from web instructional videos. In *AAAI*, 2018. [2](#)
- [129] Luowei Zhou, Yingbo Zhou, Jason J. Corso, Richard Socher, and Caiming Xiong. End-to-end dense video captioning with masked transformer. *CVPR*, pages 8739–8748, 2018. [2](#)
- [130] Luowei Zhou, Yingbo Zhou, Jason J Corso, Richard Socher, and Caiming Xiong. End-to-end dense video captioning with masked transformer. In *CVPR*, pages 8739–8748, 2018. [2](#)
- [131] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. In *arXiv preprint arXiv:1904.07850*, 2019. [4](#)
- [132] Xingyi Zhou, Anurag Arnab, Shyamal Buch, Shen Yan, Austin Myers, Xuehan Xiong, Arsha Nagrani, and Cordelia Schmid. Streaming dense video captioning. In *CVPR*, pages 18243–18252, 2024. [2](#)
- [133] Wanrong Zhu, Bo Pang, Ashish V. Thapliyal, William Yang Wang, and Radu Soricut. End-to-end dense video captioning as sequence generation. *ArXiv*, abs/2204.08121, 2022. [2](#)
- [134] Wanrong Zhu, Bo Pang, Ashish V. Thapliyal, William Yang Wang, and Radu Soricut. End-to-end dense video captioning as sequence generation. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 5651–5665, Gyeongju, Republic of Korea, 2022. International Committee on Computational Linguistics. [2](#)
- [135] 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. [4](#), [20](#)

Appendix

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 [82] 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 [16] 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.

Offline Query Enrichment w/o Annotated Intervals. In this second setup, we employ a multimodal LLaVA OneVision 72B model [39] for query enrichment as a form of training augmentation. Unlike the approach in Table 8

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 [16] 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 [39] as pre-processing, 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.

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.

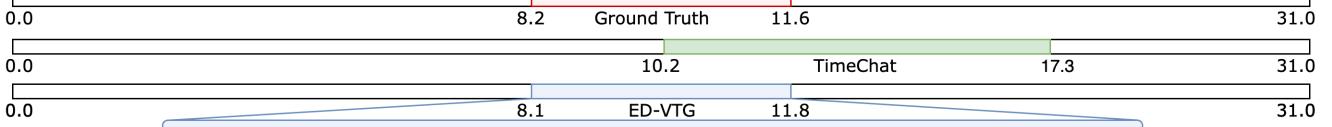
of the main paper, we do not crop the input video to the ground-truth interval in this setup. As a result, the model often incorporates irrelevant contextual information into the query, which is not helpful for localizing the desired interval. Consequently, as shown in Table B.1, this type of augmentation 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 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



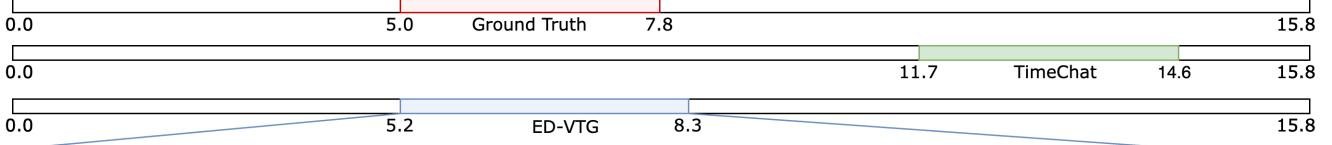
Query: person opens door.



Enriched query by ED-VTG: the person is opening the cupboard door before starting to put away groceries.



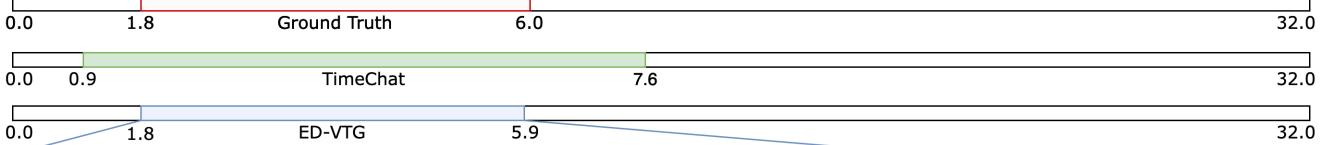
Query: the person is talking.



Enriched query by ED-VTG: the person is seen talking on their phone while standing in front of a desk with a lamp and a mirror.



Query: the person exits the room, and quickly reenters.



ED-VTG chooses to use the original query: the person exits the room, and quickly reenters.

Figure A.1. Examples of query enrichment and localization made by ED-VTG on single-query temporal grounding (STG) task from the Charades-STA [18] dataset. We also show the prediction made by one baseline model, TimeChat [82], 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.

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 ActivityNet 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.

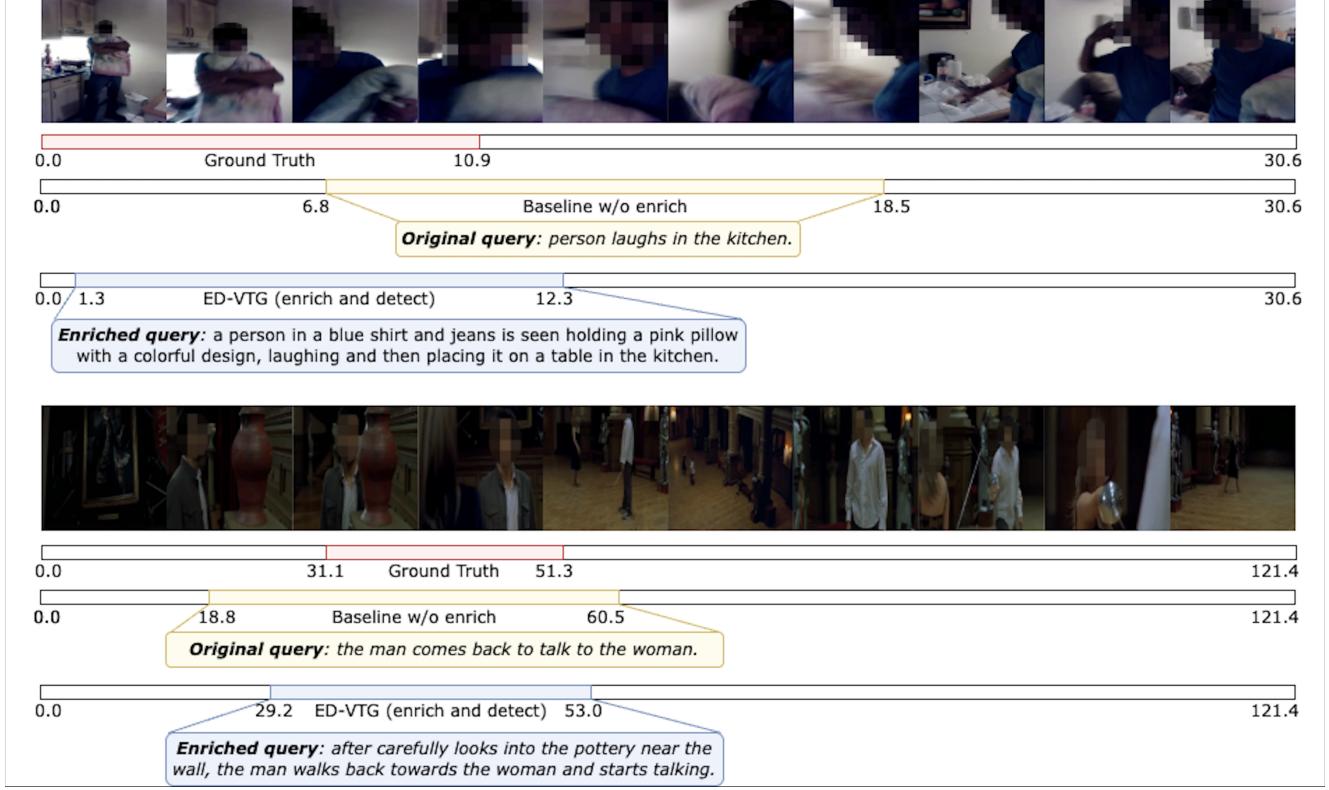


Figure A.2. **Comparison of detections of ED-VTG using its predicted enriched queries against a baseline version trained to always use the original queries.** The enriched queries contain additional relevant details and context that enable ED-VTG to perform more accurate temporal localization. In the first example, which is taken from Charades-STA [18], 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 [36] dataset, the enriched query provides additional temporal context which leads to more precise temporal boundary prediction.

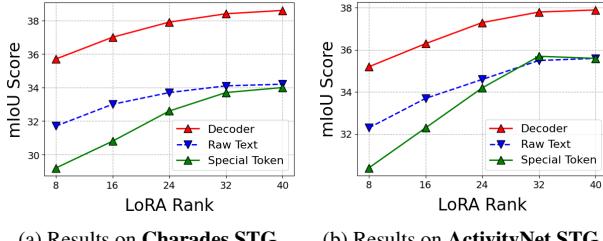


Figure B.1. **Ablation study on timestamp representation by the interval decoder.** We compare performance of our proposed lightweight decoder vs timestamp as raw text [26, 46, 57, 82] vs timestamp representation by special tokens [27, 73, 98], and find the decoder to be significantly better than both other techniques. Reported results are in FT w/o PT setting.

C. Comparison with Specialist Baselines

Table C.1 extensively compares the ED-VTG with 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 [108], UniVTG [47], MomentDiff

[44], QD-DETR [63], CG-DETR [62], etc., while models like EMB [29], EaTR [31], and SG-DETR [23] 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.

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 [39]. First, we crop the input videos between the annotated time intervals. Next, we input the original

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) [119]	X	—	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 [19]	X	—	FT	—	23.6	8.9	—	—	—	—	—	18.3	13.3	—	—		
GTR-H [9]	X	—	FT	—	62.6	39.7	—	—	50.6	29.1	—	—	40.4	30.2	—		
2D-TAN [122]	X	—	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) [124]	X	—	FT	—	56.6	36.2	—	62.1	45.5	28.3	—	42.0	33.6	22.1	—		
Moment-DETR [38]	X	236K	FT	65.8	52.1	30.6	45.5	—	—	—	—	38.0	24.7	12.0	25.5		
UMT [†] [51]	X	236K	FT	—	48.3	29.3	—	—	—	—	—	—	—	—	—	—	
UnLoc-B [108]	X	650K	FT	—	58.1	35.4	—	—	48.0	29.7	—	—	—	—	—	—	
MomentDiff [44]	X	—	FT	—	55.6	32.4	—	—	—	—	—	46.6	28.9	12.4	30.4		
LGI [64]	X	—	FT	73.0	59.5	35.5	51.4	58.5	41.5	23.1	41.1	—	—	—	—		
FlashVTG (SF+C) [10]	X	—	FT	—	60.1	38.0	—	—	—	—	—	53.7	41.8	24.7	37.6		
BAM-DETR [37]	X	—	FT	72.9	60.0	39.4	52.3	—	—	—	—	56.7	41.5	26.8	39.3		
UniVTG [47]	X	4.2M	FT	70.8	58.0	35.7	50.1	—	—	—	—	51.4	35.0	17.4	33.6		
QD-DETR (SF+C) [63]	X	—	FT	—	57.3	32.6	—	—	—	—	—	—	—	—	—		
CG-DETR (SF+C) [62]	X	—	FT	70.4	58.4	36.3	50.1	—	—	—	—	54.4	39.5	23.4	37.4		
TR-DETR (SF+C) [92]	X	—	FT	—	57.6	33.5	—	—	—	—	—	—	—	—	—		
GVL (C3D) [99]	X	—	FT	—	—	—	—	—	48.9	27.2	46.4	45.9	34.6	—	32.5		
InternVideo2* + CG-DETR [101]	X	2.1M	FT	79.7	70.0	48.9	58.8	—	—	—	—	—	—	—	—		
SG-DETR [23]	X	—	FT	—	71.1	52.8	60.7	—	—	—	—	—	46.4	33.9	42.4		
MGSL-Net [49]	X	150K	FT	—	64.0	41.0	—	—	51.9	31.4	—	42.5	32.3	—	—		
EaTR [31]	X	150K	FT	—	68.5	44.9	—	—	58.1	37.6	—	—	—	—	—		
EMB (ELA) [29]	X	—	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) [40]	✓	129M	FT	—	43.3	<u>32.6</u>	—	—	25.8	9.7	—	—	—	—	—	—	
VideoChat2 [42]	✓	2M	FT	—	—	—	—	55.5	<u>34.7</u>	17.7	38.9	—	—	—	—	—	
TimeChat [82]	✓	125K	FT	—	46.7	23.7	—	—	—	—	—	27.7	<u>15.1</u>	<u>6.4</u>	<u>18.4</u>		
HawkEye [102]	✓	715K	FT	<u>72.5</u>	<u>58.3</u>	28.8	<u>49.3</u>	<u>55.9</u>	<u>34.7</u>	<u>17.9</u>	<u>39.1</u>	—	—	—	—		
VtimeLLM [26]	✓	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		
$\Delta_{\text{Ours} - \text{HawkEye}}$	—	—	FT	5.7↑	3.8↑	6.2↑	3.3↑	11.7↑	10.4↑	4.8↑	5.8↑	—	—	—	—		
$\Delta_{\text{Ours} - \text{VTimeLLM}}$	—	—	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 [62] head for grounding tasks, using the LLM only as a video feature extractor.

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 [16], 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.

E. Example Instructions for Different Tasks

High-quality language instructions are essential for effective instruction tuning of LLMs across various downstream tasks [43, 71, 100]. For each task, we manually write one high-quality instruction as starting and generate variations

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 [39].** 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.

using GPT-4 [2]. Eventually, we manually refine the LLM-generated instructions to obtain the final version. Based on insights from M³IT [43] and TimeChat [82], 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 quantitatively - 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: DiDeMo¹ [5] 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 [97] 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: QuerYD² [68], sourced from the YouDescribe project [83], 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 COIN³ dataset [95] 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 model to localize each input query.

HiREST: The Hierarchical Retrieval and Step-captioning (HiREST)⁴ dataset [113] 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

²<https://www.robots.ox.ac.uk/~vgg/data/queryd/>

³<https://github.com/coin-dataset/annotations>

⁴<https://github.com/j-min/HiREST>

¹<https://github.com/LisaAnne/LocalizingMoments>

Task	Example Instructions
STG	<ul style="list-style-type: none"> Please look into the given video and localize the textual query: <i><Input Query></i>. 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	<ul style="list-style-type: none"> 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: <i><Input Queries></i>.
QG	<ul style="list-style-type: none"> Analyze the provided video and the question: <i><Input Question></i> 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	<ul style="list-style-type: none"> 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: <i><Input Queries></i>.

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.

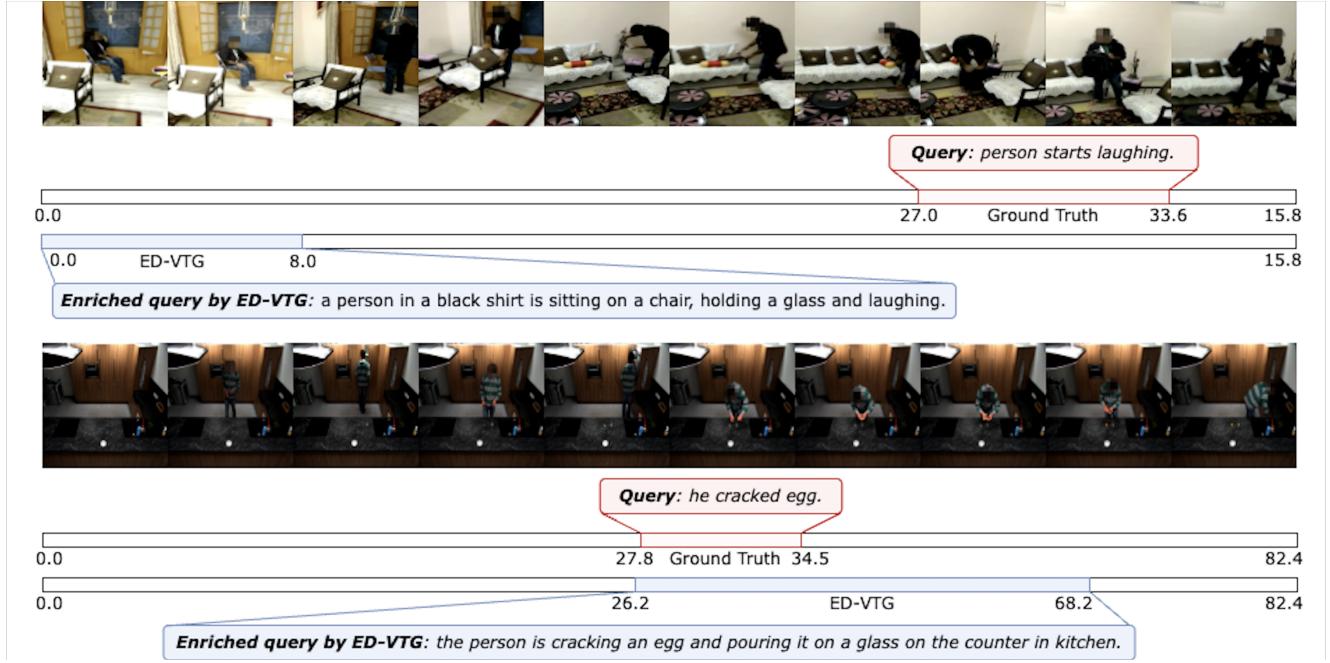


Figure F.1. **Limitations of our method.** In this figure, we show two error cases where ED-VTG fails to accurately ground the input queries. The two samples are taken from Charades-STA [18] and TACoS [80], respectively. In the first case, the model completely fails to recognize the correct interval. 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.

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.

Hyper-parameters	Notation	Value
<i>Vision Encoder</i>		
Frame encoder	–	EVA-CLIP [93]
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
<i>Interval Decoder</i>		
# Transformer layers	–	2
Transformer layer num heads	–	12
Transformer layer hidden dim	–	768
MLP dim	–	768 - 256 - 128 - 2
<i>Pre-training</i>		
Batch size	–	256
Epochs	–	40
Number of frames	–	96
Frame resolution	–	224 × 224
Max. length of text	–	2048
Loss weights	$\lambda_{LM}, \lambda_{L1}, \lambda_{gIoU}$	2, 1, 1
Optimizer	–	AdamW [53]
LoRA rank	–	32
Peak LR	–	5e-5
Warmup	–	Linear (first 8 epochs)
LR decay	–	Cosine [52]
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.

VITT: The Video Timeline Tags (VITT)⁵ [28] 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 [114] comprises 18 million narrated videos sourced from YouTube, from which we utilize the same subset as TimeChat [82]. 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

⁵<https://github.com/google-research-datasets/Video-Timeline-Tags-ViT>

weakly-supervised annotation source. The enriched queries significantly aid ED-VTG in achieving accurate grounding. Following the methodology of Vid2Seq [110], we use Whisper-timestamped [54, 77] to automatically transcribe the speech, which is then used as input queries.

CrossTask: The CrossTask⁶ [135] 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 [16] 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 [39], and manually verify a small portion (5%) of the filtered negative queries for quality assurance.

VideoCC: VideoCC⁷ [65] 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.

H.2. Fine-tuning and Evaluation Datasets

Charades-STA: Charades-STA⁸ [18] 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 [44, 47, 63, 92], 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.

⁶<https://github.com/DmZhukov/CrossTask>

⁷<https://github.com/google-research-datasets/videoCC-data>

⁸<https://github.com/jiyanggao/TALL>

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 [18]	32	120	24	96	2	1	1	3e-5	1e-5	1e-5
	ActivityNet-Captions [36]	32	30	6	144	1	1	1	3e-5	1e-5	1e-5
	TACoS [80]	32	120	24	144	4	1	1	3e-5	1e-5	1e-5
STG	Charades-CD-OOD [112]	32	120	24	96	2	1	1	3e-5	1e-5	1e-5
	ActivityNet-Captions [36]	32	30	6	144	3	1	1	3e-5	1e-5	1e-5
	TACoS [80]	32	120	24	144	4	1	1	3e-5	1e-5	1e-5
	YouCook2 [127]	32	120	24	144	1	1	1	3e-5	1e-5	1e-5
AG	HT-Step [3]	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 [107] 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.

Charades-CD-OOD: Charades-CD-OOD⁹ [112] 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-Captions¹⁰ [36] dataset is a comprehensive resource designed for dense video captioning and temporal localization tasks, derived from the original ActivityNet [36] 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 [7, 9, 26, 32, 48, 82, 103], we use the val_2 for evaluation. We report both STG and VPG performance of ED-VTG on ActivityNet-Captions.

TACoS: The TACoS¹¹ [80] dataset is a specialized col-

lection derived from the MPII Cooking Composite Activities video corpus [88], 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 YouCook2¹² [127] 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 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-GQA¹³ [107] 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 [105] 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

⁹https://github.com/ytzsy/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>

¹³<https://github.com/doc-doc/NExT-GQA>

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-Step¹⁴ [3] 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 [61] 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 [35], which include rich descriptions of each step. Since HTStep releases the negative queries, we report article grounding (AG) performance on this dataset.

¹⁴<https://github.com/facebookresearch/htstep>