Decompose and Conquer: Compositional Reasoning for Zero-Shot Temporal Action Localization

Haoyu Tang[1], Tianyuan Liang[1], Han Jiang[2], Xuesong Liu[3], Qinghai Zheng[4], Yupeng Hu[1]*

[1] School of Software, Shandong University
[2] Xi'an Jiaotong University
[3] James Watt School of Engineering, University of Glasgow
[4] College of Computer and Data Science, Fuzhou University
tanghao258@sdu.edu.cn, liangtianyuan@mail.sdu.edu.cn, jh.01@stu.xjtu.edu.cn, x.liu.10@research.gla.ac.uk, zhengqinghai@fzu.edu.cn, huyupeng@sdu.edu.cn

Abstract

Current Zero-Shot Temporal Action Localization (ZSTAL) methods, whether training-based or training-free ones, still predominantly rely on a single, unified query to localize an entire action. This unified representation is fundamentally ill-suited for complex real-world activities, as it fails to capture their internal compositional structure and adapt to dynamic, multi-stage variations across videos. To address this, we regard ZSTAL as a compositional reasoning task and introduce CASCADE, a Context-Aware Staged Action DEcomposition framework. Inspired by the human cognitive process of perceiving context, decomposing events, and reconstructing instances, CASCADE follows a training-free pipeline. It first perceives the video's context by leveraging a Multimodal Large Language Model (MLLM) to both filter out irrelevant actions and then generate a rich, video-specific caption for each action present in the video. An LLM then decomposes this caption into multiple, temporally ordered stages, which serve as fine-grained queries to guide the MLLM in estimating frame-level confidence scores. Recognizing that this decomposition can fragment a single action, a novel hierarchical merging logic then reconstructs complete instances by intelligently fusing these preliminary temporal segments based on their semantic progression and coherence. Extensive experiments and ablation studies on THUMOS14 and ActivityNet-1.3 show that CASCADE not only sets a new state-of-the-art among training-free methods but, most notably, significantly outperforms all prior training-based approaches on ActivityNet-1.3.

Introduction

The increasing prevalence of surveillance devices in domains like public security and sports analytics has driven the demand for advanced video understanding techniques. A key task in this area is Temporal Action Localization (TAL) (Ju et al. 2022; Nag et al. 2022b; Phan et al. 2024), which aims to identify the specific category and the precise temporal boundaries of actions in untrimmed videos. Despite its potential, current TAL methods (Shao et al. 2023; Shi et al. 2023a,b; Zhang et al. 2022; Zhao et al. 2017) mainly follows the supervised or weakly-supervised paradigms, which face two critical limitations. First, creating either fine-grained, frame-level boundaries or coarse, video-level labels is notoriously labor-intensive and time-consuming. Second, these methods learn features overly specific to their training data, causing them to fail on out-of-distribution (OOD) or "in-the-wild" videos and thus limiting their real-world utility. To this end, the community has shifted towards Zero-Shot Temporal Action Localization (ZSTAL) (Bosetti et al. 2024; Ju et al. 2022), a new paradigm where test-time action categories are entirely unseen during training. Based on the impressive performance of MLLMs on visual understanding and reasoning (Maaz et al. 2023; Lin et al. 2023; Zhan et al. 2025), current ZSTAL methods, whether training-based (Ju et al. 2022) or training-free (Han et al. 2025; Aklilu, Wang, and Yeung-Levy 2024), predominantly treat actions as unified, indivisible events. They typically rely on static textual queries—such as a prompt with class label (e.g., a video of "[action]") or a general, one-paragraph description—to score frame-level relevance. This unified, indivisible representation creates two critical problems. 1) **Failure to Model Action Structure.** As shown in Figure 1, this strategy treats a complex action "baking cookies" as a single event, ignoring the crucial sequence and varying importance of its internal stages for action localization (e.g., 'cutting chocolate', 'mixing in-

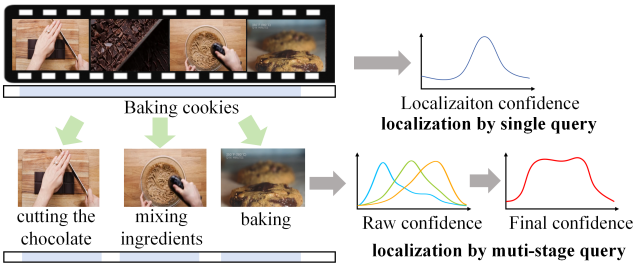


Figure 1: Illustration of a complex action "Baking cookies", which involves multiple stages (e.g., cutting chocolate, mixing, baking). Our stage-aware localization process clearly outperforms the single query result.

*Corresponding author.

gredients', 'baking'). 2) **Lack of Adaptability to Cross-Scene Dynamics.** A same action may appear with different stages or durations across different videos. For example, in another video, 'cutting chocolate' stage may be omitted, or stages like "Preparing Ingredients" and "Melting Chocolate" may be included. All these dynamic changes make a static query brittle and unreliable. How to address these problems? In this paper, we argue that the key to robust ZSTAL lies in abandoning unified and static queries. Instead, we embrace a compositional reasoning approach, inspired by how humans comprehend complex events by **perceiving the global context**, **decomposing the event into multiple stages**, and finally **reconstructing the action instances**. Based on this philosophy, we propose a **C**ontext-**A**ware **S**taged **A**ction **DE**composition model, CASCADE, for ZSTAL. Instead of treating an action as an indivisible unit, CASCADE begins with a broad perception of the global context, leveraging an MLLM to filter the set of relevant actions. Transitioning from this coarse-grained perception to a finer understanding, the stage-aware decomposition is then initiated. This module begins with a crucial perceptual step: generating a rich, video-specific caption that details the action's unique dynamics in the video. This caption provides the necessary foundation for the subsequent parsing into a structured sequence of key and non-key stages. Subsequently, CASCADE grounds each stage via stage-wise confidence estimation. Finally, to reconstruct complete action instances for action localization, our CASCADE presents a hierarchical merging logic to intelligently fuse the localized stage segments, which respects both the semantic progression and temporal coherence of stages. This entire pipeline is training-free, operating solely with off-the-shelf models. Our main contributions are summarized as follows:

- We propose **CASCADE**, a new training-free paradigm for ZSTAL that shifts from unified query matching to compositional action reasoning, effectively harnessing modern MLLMs to parse and merge complex actions.
- We design two core components in our pipeline: a stage-aware decomposition module to parse action structure into stages, and a compositional action reconstruction module that fuses stage segments into coherent action instances with a novel hierarchical merging logic.
- We demonstrate that CASCADE sets a new state-of-the-art among training-free methods on THUMOS14 and ActivityNet-1.3. Most notably, on ActivityNet-1.3, our training-free approach significantly outperforms all prior training-based ZSTAL methods.

Related Work

Temporal Action Localization

Temporal action localization aims to localize and classify action instances within a video. Existing TAL methods mainly follow the supervised (frame-level annotations) or weakly supervised settings (video-level annotations). The methods under these two settings can be broadly classified into three categories: 1) Two-stage methods (Ju et al. 2022; Qing et al. 2021; Xia et al. 2022; Zhao, Thabet, and Ghanem 2021; Zhao, Wang, and Zhao 2023) first generate class-agnostic action proposals and then classify each proposal into an action category. 2) One-stage methods (Liu et al. 2024c; Nag et al. 2022a; Shao et al. 2023; Shi et al. 2023a; Zhang, Wu, and Li 2022) perform the action proposal generation and classification in a unified, end-to-end manner, with popular techniques like contrastive learning to refine the representations of video snippets. 3) Query-based methods (Aklilu, Wang, and Yeung-Levy 2024; Kim et al. 2024; Kim, Lee, and Heo 2023; Zhu et al. 2024) adopted the DETR settings to employ a set of learnable queries to interact with video features for the set prediction of actions. Despite their great progress, the reliance of these methods on a closed-set assumption and substantial labeled data restricts their adaptability in real-world scenarios.

Zero-shot Temporal Action Localization

To overcome the closed-set limitation, ZSTAL has been introduced, which aims to localize actions from unseen categories. The field is currently dominated by two main branches: training-based and training-free. Training-based ZSTAL methods adapt a model on a dataset of seen action classes with the goal of generalizing to unseen ones. These approaches often involve fine-tuning components of a VLM to better align visual and textual representations for TAL. For instance, some methods focus on designing efficient tuning strategies like prompt learning (Ju et al. 2022), while others work on decoupling the localization and classification tasks to improve both boundary precision and semantic alignment (Li et al. 2024). While these methods have achieved impressive results on benchmark datasets, their reliance on labeled data for seen classes makes them data-dependent and vulnerable to performance degradation when faced with great domain shifts. Training-free ZSTAL methods represent a more recent and flexible paradigm. Without any task-specific training, these methods often leverage the rich knowledge of pre-trained VLMs for action inference (Han et al. 2025; Aklilu, Wang, and Yeung-Levy 2024). The common blueprint for these methods involves the following steps: 1) computing frame-wise similarity scores between video features and text prompts describing target actions; 2) using thresholding or ranking mechanisms to extract the final action segments. Crucially, the reliance on static queries to represent actions as unified entities prevents these methods from robustly handling the compositional structure and dynamic variations of real-world activities.

MLLMs for Video Understanding

Multimodal Large Language Models have recently driven significant progress in video understanding, which have excelled in high-level semantic tasks; for instance, models like Video-ChatGPT (Maaz et al. 2023) and Video-LLaVA (Lin et al. 2023) enable accurate video question answering, while others (Li et al. 2023) generate coherent textual summaries of complex events. In the context of temporal localization, recent studies have demonstrated the potential of MLLMs to ground events without task-specific training (Achiam et al.

2023; Wang et al. 2022). Some approaches explicitly prompt MLLMs with temporal tokens to directly predict action boundaries (Chen et al. 2024; Wang et al. 2024; Wu et al. 2025). While promising, these methods tend to treat actions as single, indivisible events, overlooking the complex, multi-stage nature of real-world activities. This simplification limits their ability to accurately locate actions with rich internal structure. Similar structured reasoning ideas also appear in interest-aware message passing models(Liu et al. 2021), which aim to enhance fine-grained representation learning. In contrast, our framework leverages the reasoning capabilities of MLLMs precisely to decompose actions into their constituent stages, enabling a more granular and accurate localization process.

Our Proposed Framework

Preliminary

Given a video $V \in \mathbb{R}^{3\times H\times W\times T}$, where T is the number of frames, and a predefined action set $\mathcal{A} = \{a_1, a_2, \ldots, a_N\}$ with N action classes, the goal of TAL is to localize all the action instances $Seg_j = (\tau_j^s, \tau_j^e, a_n)$, where τ_j^s and τ_j^e are the start and end times (satisfying $1 \le \tau_j^s \le \tau_j^e \le T$) of j-th instance, and $a_n \in \mathcal{A}$ is the corresponding action class. Note that each video may contain multiple action instances from different categories or repeated actions. The proposed framework is illustrated in Figure 2.

Context-Guided Action Filtering

The first step of our framework is to find the set of actions $\hat{\mathcal{A}}$ occurred in the video V from the predefined action set $\mathcal{A}$. Conventional ZS-TAL methods often tackle this by aligning video features with textual action descriptions with a pre-trained CLIP model. This approach, however, inevitably introduces additional redundancy by requiring a large CLIP in addition to the MLLM used for localization. Therefore, our model instead directly prompts the MLLM, explicitly instructing it to recognize and output only the actions present in the video. Formally, this process is defined as follows:

$$\hat{\mathcal{A}} = \text{MLLM}_{\text{detect}}(V, \mathcal{A}; \tau) \tag{1}$$

where τ is a threshold that filters out actions with lower confidence. Subsequently, for each identified action $\hat{a} \in \hat{\mathcal{A}}$, our framework proceeds to localize its precise temporal boundaries.

Stage-Aware Decomposition

Given an action $\hat{a} \in \hat{\mathcal{A}}$, a detailed description is needed to guide its localization. A simple class label is often insufficient to describe the complex and varied manifestations of an action in a video. While existing methods might use a LLM to generate a generic text description from the action label, these descriptions ignore crucial video-specific variations. For instance, a generic description for "playing basketball" is too broad to be effective, as it fails to distinguish between a video depicting a slam dunk and another showing a three-point shot. To address this, our method provides both the video V and the action label $\hat{a}$ to the MLLM, prompting it to generate a context-specific description $Cap_{\hat{a}}$ that is grounded in the action's specific occurrence in the video. This process is expressed as:

$$Cap_{\hat{a}} = \text{MLLM}_{\text{desc}}(V, \hat{a}) \tag{2}$$

where desc denotes the prompt template designed to elicit this detailed, context-specific description. However, while this caption is context-aware, treats the action as a monolithic block, which presents two limitations. First, it obscures the action's inherent temporal structure—the distinct sequence of stages that define its progression. For instance, "shooting a goal" is not a single event but a sequence of 'running', 'swinging the leg', and 'kicking the ball'. Second, this approach fails to account for the varying importance of these stages. The stage 'kicking the ball' is a definitive indicator of the action, making it a key stage. In contrast, 'running' is ambiguous; it could be part of numerous other activities and is only relevant when temporally connected to the key shooting motion. To capture both this temporal structure and stage importance simultaneously, our framework employs the LLM to perform a stage-aware decomposition of the caption $Cap_{\hat{a}}$. The LLM is prompted to jointly output a sequence of annotated tuples, where each contains both a sub-caption that describes the current action stage and a stage indicator. This process is formalized as:

$$\{(k_i, s_i)\}_{i=1}^{M} = \text{LLM}\left(Cap_{\hat{a}}\right) \tag{3}$$

where k_i denotes the stage indicator of the i-th sub-caption s_i, and M is the number of stages. The indicator of all non-key stages are assigned $k_i = 0$, while for a key stage, we assign a positive integer ($k_i > 0$) that enumerates its temporal order relative to other key stages.

Stage-wise Confidence Estimation

In the next, our framework recognizes the segments of all stages of the action $\hat{a}$ based on the sub-captions and corresponding stage indicators $\{(k_i, s_i)\}_{i=1}^{M}$. Inspired by prior work (Aklilu, Wang, and Yeung-Levy 2024), we adopt a confidence estimation approach that queries the MLLM to assess whether each frame's visual content aligns with the sub-caption s_i of i-th stage. Specifically, for each frame F_t, the MLLM generates binary logits for the i-th stage as follows:

$$l_{t,i}^{y}, l_{t,i}^{n} = \text{MLLM}\left(F_t, s_i, \hat{a}\right) \tag{4}$$

where $l_{t,i}^{y}$ and $l_{t,i}^{n}$ are the logits corresponding to the "yes" and "no" tokens in the MLLM's output vocabulary, respectively. These values represent the model's confidence for the affirmative and negative assessment of whether the frame belongs to the i-th stage. This formulation, which includes the overall action $\hat{a}$ as context, encourages the MLLM to ground its understanding of each stage in the corresponding visual evidence, thereby precisely evaluating the stage-specific visual context. Subsequently, these logits are normalized to produce a soft confidence as:

$$p_{t,i} = e^{l_{t,i}^{y}} / (e^{l_{t,i}^{y}} + e^{l_{t,i}^{n}}) \tag{5}$$

where $p_{t,i}$ indicates the likelihood that frame F_t represents the i-th stage. To enhance computational efficiency,

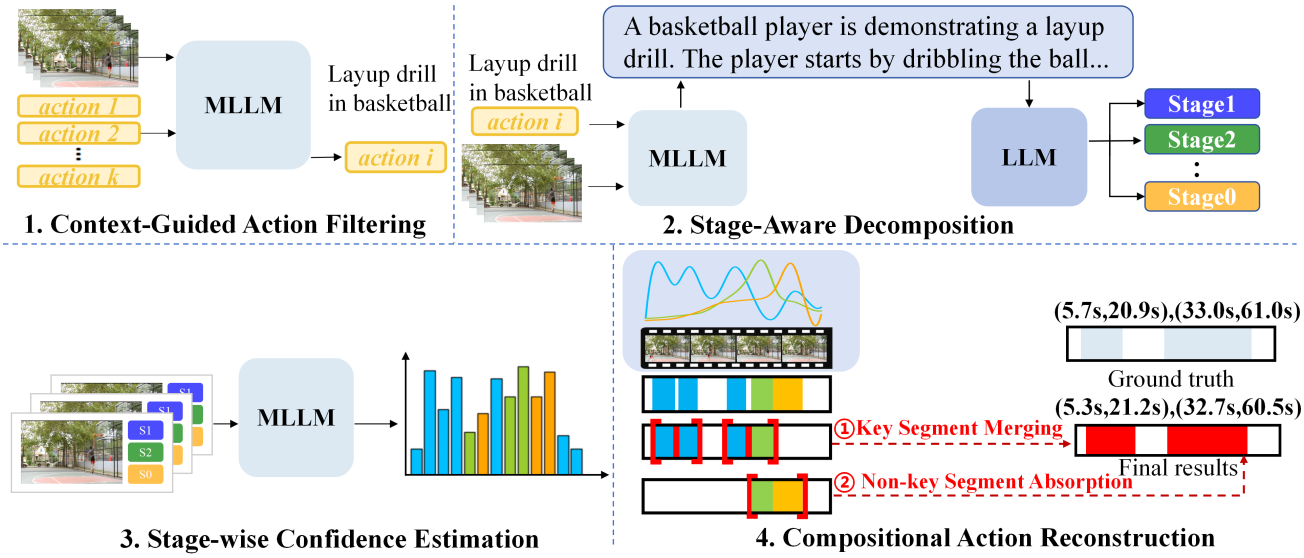


Figure 2: Overview of our framework. 1) Context-Guided Action Filtering: MLLM infers all potential action categories occurred in a video. 2) Stage-Aware Decomposition: The video and inferred actions are used to generate a detailed caption, which is then decomposed into sequential stages. 3) Stage-wise Confidence Estimation: MLLM computes frame-level confidence for each stage. 4) Compositional Action Reconstruction: Confidence scores are aggregated into raw segments and then merged together to obtain the final action instances.

it is impractical to query the MLLM sequentially for every stage in every frame. Instead, we formulate a single prompt to evaluate all M stages for a given frame concurrently, thereby obtaining the logits for all stages in a single forward pass. Details of this batched prompt are in the Appendix. This process yields a set of stage-wise confidence $\{p_{t,1}, p_{t,2} \ldots, p_{t,M}\}$ for each frame F_t. To derive a single, representative label for the frame, we select the stage with the maximum confidence. This process is defined as:

$$i^* = \arg \max_{i \in 1,\ldots,M} p_{t,i}, \tag{6}$$

$$p_t = p_{t,i^*}, \quad \hat{k}_t = k_{i^*} \tag{7}$$

where p_t represents the peak confidence value for the frame. $\hat{k}_t$ is the indicator of that winning stage, which provides the semantic label of the stage that produced this peak score. This detailed, frame-by-frame annotation is foundational for constructing coherent action instances.

Compositional Action Reconstruction

In the next, our framework proceeds to reconstruct the final action instances through a hierarchical merging process. This process begins by identifying contiguous stages of high-confidence frames, which are then merged together based on the temporal consistency.

Contiguous Stages Localization Given the peak confidence $\{p_1, p_2 \ldots, p_T\}$ of all frames, we generate a set of raw segments by aggregating contiguous frame sequences where the confidence p_t exceeds a predefined threshold θ. These high-confidence segments are identified as foreground (part of an action stage), which are then divided into two segment sets based on their constituent frames' stage indicators as follows:

$$S^{key} = \left\{ (t^s, t^e) \;\middle|\; \forall t \in [t^s, t^e],\ p_t > \theta,\ \hat{k}_t > 0 \right\} \tag{8}$$

$$S^{non} = \left\{ (t^s, t^e) \;\middle|\; \forall t \in [t^s, t^e],\ p_t > \theta,\ \hat{k}_t = 0 \right\} \tag{9}$$

where the stage indicators $\hat{k}_t$ of each frames in key segment set S^{key} and non-key segment set S^{non} are positive or zero, respectively. After all key segments are obtained, we must assign a stable, representative stage indicator to each key segment, so that the final temporal merging of action instances can be facilitated. For the o-th key segment $(t_o^s, t_o^e) \in S^{key}$, we determine its segment-level stage indicator k_o^{seg} by performing a majority vote over the stage indicators of its constituent frames. This strategy ensures that the segment's label is robust to sporadic frame-level misclassifications. The segment-level stage indicator k_o^{seg} is computed as:

$$k_o^{seg} = \arg\max_m \sum_{t=t_o^s}^{t_o^e} \mathbb{I}(\hat{k}_t = m) \tag{10}$$

where $\mathbb{I}(\cdot)$ is the indicator function, which means that its output equals 1 when $\hat{k}_t = m$, and 0 otherwise. At this point, the video has been transformed from a sequence of frame-level predictions into a collection of semantically labeled key segments and auxiliary non-key segments, paving the way for the final sequential stage merging logic.

Sequential Stage Merging This section introduces the details of merging these segments into semantically and temporally coherent action instances. The process unfolds in

two phases: first, absorbing adjacent non-key segments, and second, composing complete actions by merging key segments based on their logical stage progression. (1) **Non-key Segment Absorption**: Non-key segments often represent transitional or auxiliary phases that provide crucial context to the main action. Therefore, we merge a non-key segment into an adjacent key segment to form a more complete initial instance. For a non-key segment $(t_b^s, t_b^e) \in S^{non}$ and a key segment $(t_o^s, t_o^e) \in S^{key}$, they are considered temporally adjacent if $t_b^e = t_o^s$ or $t_o^e = t_b^s$. If so, the two segments are merged, and the non-key segment is absorbed. The new merged key segment will be defined as:

$$(t_o^s, t_o^e) = (\min(t_b^s, t_o^s),\ \max(t_b^e, t_o^e)) \tag{11}$$

The merged segment inherits the stage indicator k_o^{seg} from the original key segment. (2) **Key Segment Merging**: The merging of key segment is governed by a principled merging logic that respects both the semantic progression (stage order) and temporal coherence of the action. Let $(t_o^s, t_o^e, k_o^{seg})$ and $(t_{o+1}^s, t_{o+1}^e, k_{o+1}^{seg})$ be two temporally consecutive key segments from S^{key}. We apply two merging rules: **Case 1: Sequential Stage Indicators (SI)** $(k_o^{seg} < k_{o+1}^{seg})$**.** The strict increasing order of the stage indicators signifies a logical progression between two distinct stages of the same action instance. We merge them to reconstruct this semantic flow, forming a longer, multi-stage key segment, which is $(t_o^s,\ t_{o+1}^e, k_{o+1}^{seg})$. The merged segment adopts the indicator of the latter stage, representing the action's advancement. **Case 2: Identical Stage Indicators (II)** $(k_o^{seg} = k_{o+1}^{seg})$**.** The same stage indicator of two consecutive key segments may suggest either 1) the start of a new action instances; or 2)a single continuous stage may have been fragmented by prediction noise. To differ between them, we merge them if they are sufficiently close in time. We measure this proximity using a temporal density ρ_t, defined as the ratio of the segments' combined duration to their total span:

$$\rho_t = \frac{(t_o^e - t_o^s) + (t_{o+1}^e - t_{o+1}^s)}{t_{o+1}^e - t_o^s}. \tag{12}$$

If ρ_t exceeds a density threshold θ_t, the two segments are considered part of the same action that should be merged as a new key segment, which is $(t_o^s,\ t_{o+1}^e, k_o^{seg})$. This rule effectively repairs fragmented detections of a single stage while preserving its stage identity. This entire merging process is performed iteratively until no more segments can be merged. The final set of segments in S^{key} constitutes the complete, localized action instances $Seg_j = (\tau_j^s, \tau_j^e)$ for action $\hat{a}$.

Experiment

Datasets and Metrics

We evaluate our method on ActivityNet-1.3 (Heilbron et al. 2019) and THUMOS14 (Idrees et al. 2017). ActivityNet-1.3 includes 200 action classes and 19,994 videos, with 10,024 for training, 4,926 for validation, and 5,044 for testing. THUMOS14 contains 20 sports-related actions, comprising 200 training videos and 213 test videos. Following the standard protocols (Ju et al. 2022; Liberatori et al. 2024; Nag et al. 2022b) of ZSTAL, we adopt two data split settings: 75%/25% and 50%/50%. These splits define the set of unseen classes reserved exclusively for testing. Crucially, as our method is training-free, it does not utilize the training set for any parameter updates. Our evaluation is performed solely on the testing set containing these held-out unseen classes. To ensure statistical robustness, we report the average performance over 10 random data splits for each setting. For performance evaluation, we report the standard mean Average Precision (mAP). Following convention, we calculate mAP at various temporal Intersection over Union (tIoU) thresholds: {0.3, 0.4, 0.5, 0.6, 0.7} for THUMOS14 and {0.5, 0.75, 0.95} for ActivityNet-1.3.

Implementation Details

To comprehensively verify the performance, our framework deploys two different pretrained models, Qwen-2.5-VL-7B (Bai et al. 2025) and LLaVA-1.5-7B(Liu et al. 2024b), as the backbone MLLM. Due to resource limitations, we did not deploy larger-scale MLLMs. Even so, our framework empowers these 7B models to yield excellent localization results, underscoring the efficacy of our approach. Besides, DeepSeek-V3 (Liu et al. 2024a) is adopted to process video captions for stage-aware decomposition. θ and θ_t in our CASCADE framework are all set to 0.9. More details of prompts are in the Appendix. All experiments are conducted on a 80GB NVIDIA A100 GPU.

Performance Comparison

Our framework is compared with several state-of-the-art baselines, including training-based ZSTAL approaches (e.g., Eff-Prompt (Ju et al. 2022), STALE (Nag et al. 2022b), DeTAL (Li et al. 2024)), and training-free methods approaches (e.g., T3AL (Liberatori et al. 2024), FreeZAD (Han et al. 2025), ZEAL (Aklilu, Wang, and Yeung-Levy 2024)). As illustrated in Table 1, results for training-based and training-free methods are distinguished by white and gray backgrounds, respectively, and the best results are highlighted in bold. On THUMOS14 dataset, CASCADE consistently outperforms all existing training-free baselines. Under the 75%/25% split, our CASCADE-LLaVA variant achieves a mean Average Precision (mAP) of 13.7%, surpassing the previous best training-free method, ZEAL, by a margin of 2.1%. In the more challenging 50%/50% split, our CASCADE-LLaVA variant leads with 12.2% mAP, marking a 1.5% improvement over the same baseline. These results validate the robustness of CASCADE in accurately localizing unseen actions without relying on any domain-specific training data. The superiority of our framework is even more pronounced on the larger and more diverse ActivityNet-1.3 dataset. Remarkably, our training-free CASCADE surpasses not only its direct competitors but also all training-based ZSTAL methods that leverage supervision from seen categories. Under the 75%/25% split, CASCADE-LLaVA sets a new SOTA with 32.6% mAP. This represents a substantial absolute improvement of 14.3% over the best-performing training-free method, FreeZAD, and a significant 7.0% gain over the top-performing training-based method, DeTAL.

Settings	Methods	Label	THUMOS14						ActivityNet-1.3			
			0.3	0.4	0.5	0.6	0.7	mAP	0.5	0.75	0.95	mAP
Fully supervised	ActionFormer	✓	82.1	77.8	71.0	59.4	43.9	66.8	53.5	36.2	8.2	35.6
Zero-shot	Eff-Prompt	✓	39.7	31.6	23.0	14.9	7.5	23.3	37.6	22.9	3.8	23.1
	STALE	✓	40.5	32.3	23.5	15.3	7.6	23.8	38.2	25.2	6.0	24.9
	DeTAL	✓	39.8	33.6	25.9	17.4	9.9	25.3	39.3	26.4	5.0	25.8
75% Seen	T3AL	×	19.2	12.7	7.4	4.4	2.2	9.2	28.1	14.9	3.3	15.4
	FreeZAD	×	21.2	13.6	8.3	4.7	2.5	10.0	33.5	17.5	3.9	18.3
25% Unseen	ZEAL	×	22.1	16.1	11.0	5.7	3.0	11.6	-	-	-	-
	CASCADE-Qwen	×	**23.9**	17.5	11.7	**7.6**	4.3	13.0	41.4	27.4	7.1	25.3
	CASCADE-LLaVA	×	23.8	**17.9**	**14.0**	**7.6**	**5.1**	**13.7**	**52.7**	**36.7**	**8.3**	**32.6**
Zero-shot	Eff-Prompt	✓	37.2	29.6	21.6	14.0	7.2	21.9	32.0	19.3	2.9	19.6
	STALE	✓	38.3	30.7	21.2	13.8	7.0	22.2	32.1	20.7	5.9	20.5
	DeTAL	✓	38.3	32.3	24.4	16.3	9.0	24.1	34.4	23.0	4.0	22.4
50% Seen	T3AL	×	20.7	14.3	8.9	5.3	2.7	10.4	25.8	13.9	3.1	14.3
	FreeZAD	×	20.7	13.4	8.3	4.7	2.5	9.9	34.1	17.9	4.0	18.7
50% Unseen	ZEAL	×	21.1	15.0	9.9	5.0	2.6	10.7	-	-	-	-
	CASCADE-Qwen	×	21.6	15.3	10.2	6.4	3.8	11.5	41.1	27.5	7.4	25.3
	CASCADE-LLaVA	×	**22.4**	**16.1**	**11.8**	**6.5**	**4.2**	**12.2**	**52.1**	**36.5**	**8.4**	**32.3**

Table 1: Performance Comparisons on THUMOS14 and ActivityNet-1.3. We also include a supervised method for reference.

Method	**mAP @ IoU**			**mAP**
	0.5	**0.75**	**0.95**	
MLLMs with $\mathcal{A}$	8.9	3.1	0.4	4.1
MLLMs with $\hat{\mathcal{A}}$	9.2	3.4	0.5	4.4
MLLMs with $Cap_{\hat{a}}$	6.8	2.1	0.2	3.0
CASCADE-Qwen	41.4	27.4	7.1	25.3

Table 2: Ablations of MLLM performance with different inputs on ActivityNet-1.3.

Identical Indicator	**Sequential Indicator**	**mAP @ IoU**			**mAP**
		0.5	**0.75**	**0.95**	
✓	✓	41.4	27.4	7.1	25.3
✓	×	31.6	20.2	5.3	19.0
×	✓	21.6	10.5	2.7	11.6
×	×	18.8	9.5	2.5	10.3

Table 3: Ablations of the Merging of Identical Stage Indicator and Sequential Stage Indicator on ActivityNet-1.3.

The performance gap further widens in the 50%/50% setting, where CASCADE-LLaVA achieves 32.3% mAP, outperforming FreeZAD and DeTAL by 13.6% and 9.9%, respectively. This outstanding performance, achieved without any annotations, highlights the exceptional effectiveness and generalization capability of our proposed design, establishing a new benchmark for ZSTAL.

Ablation Studies

Impact of Different MLLM Input. As shown in Table 2, we ablate the MLLM query formulation to validate our stage-aware design. Using the full action set $\mathcal{A}$ or the filtered list $\hat{\mathcal{A}}$ from Context-Guided Action Filtering as direct queries yields poor performance (4.1% and 4.4% mAP respectively), indicating simple labels are insufficient. Notably, using the video-specific caption $Cap_{\hat{a}}$ from Stage-aware Decomposition as a query further degrades performance to 3.0% mAP. This confirms our hypothesis that a monolithic description, despite its detail, is an ineffective query as it obscures the action's key stages. In stark contrast, our full CASCADE framework, by decomposing the action into explicit stages, achieves a remarkable 25.3% mAP, highlighting that our stage-aware decomposition and merging is critical for precise localization.

Impact of Key Segment Merging. As shown in Table 3, under a merging threshold of $\theta_t = 0.9$ with CASCADE-Qwen, our full merging strategy achieves a 25.3% mAP. We analyze the contribution of its two components: merging segments with identical and sequential stage indicators. Disabling sequential indicator merging reduces performance to 19.0% mAP, whereas disabling identical indicator merging results in a sharper drop to 11.6%, only slightly above the 10.3% no-merge baseline. This confirms both rules are critical, but merging identical-stage fragments is more crucial, as it first repairs prediction noise to create stable segments before they can be linked in a logical sequence. Due to the page limits, more ablation studies of merging strategies are in Appendix.

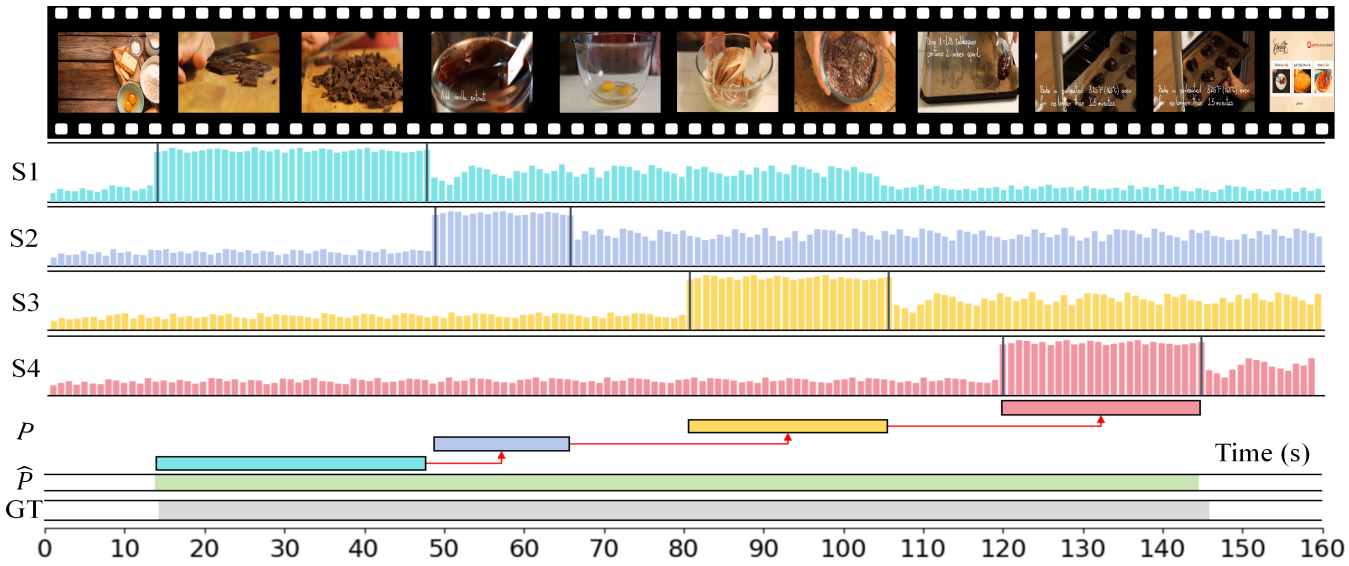
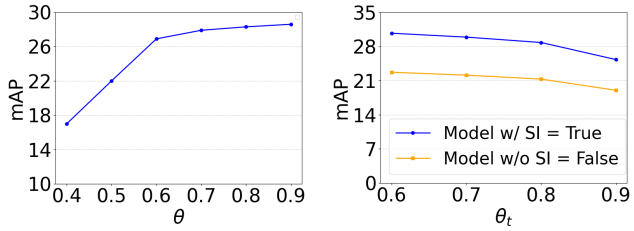


Figure 3: Visualization of "Baking cookies" of a video in ActivityNet-1.3. Si denotes the i-th stage, P is obtained by thresholding with θ on the confidence of the four stages. $\hat{P}$ denotes the final merged result. GT denotes the ground-truth.



(a) Impact of θ (b) Impact of θ_t

Figure 4: Analysis of θ and θ_t on ActivityNet-1.3.

Impact of θ and θ_t. We analyze the impact of the confidence threshold θ in Figure 4a. At low values, θ causes a significant drop in mAP, as excessive background and noisy frames are erroneously included in the stage segments. As θ increases, performance improves considerably, peaking at θ=0.9. This effectiveness at a high threshold is due to the MLLM's confident predictions, where the output logits for "yes" and "no" are highly polarized. Consequently, a strict θ=0.9 serves as an effective noise filter while preserving the majority of crucial, high-confidence frames, leading to the best localization performance. For the density threshold θ_t which governs the merging of same-stage segments, the results are presented in Figure 4b. The analysis confirms the importance of our sequential stage merging rule (blue line), which consistently outperforms the ablated version (orange line). We also observe that performance is highest with a moderately inclusive threshold and degrades as θ_t becomes overly strict (e.g., 0.9), which can prevent the model from repairing valid fragments. However, our experiments on THUMOS14 revealed that a stricter θ_t=0.9 achieved the best results. To maintain a single, robust hyperparameter across both benchmarks, we selected θ_t =0.9 as our default. This represents a trade-off, optimizing for peak performance on THUMOS14 while retaining highly competitive results on ActivityNet-1.3.

Qualitative Results.

Figure 3 visualizes our localization process for the "Baking cookies" action. Our framework first decomposes the action into four semantic stages (S1:"Preparing Ingredients", S2: "Mixing Ingredients", S3: "Melting Chocolate", S4:"Baking") and computes their frame-wise confidence scores. While initial thresholding of these scores yields a set of temporally disjoint raw proposals (P), our compositional assembly logic successfully fuses these fragments. The resulting merged instance ($\hat{P}$) reconstructs the complete action narrative by bridging the semantic gaps between stages, leading to a final prediction that aligns remarkably well with the ground truth (GT). More visualization results can be found in Appendix.

Conclusion

In this paper, we proposed CASCADE, a novel training-free framework that regards the temporal action localization as a compositional reasoning task. By decomposing actions into semantic stages and then merging them with a hierarchical merging logic, CASCADE not only establishes a new state-of-the-art among training-free methods but, most notably, surpasses all prior training-based competitors on the challenging ActivityNet-1.3 dataset. This success highlights a promising new direction where the explicit compositional reasoning of large language models can serve as a powerful alternative to traditional supervised feature learning. Ultimately, our results demonstrate that abandoning unified representations in favor of a granular, stage-aware analysis is a more effective and generalizable strategy for complex temporal action localization.

Acknowledgments

This work was supported in part by the National Natural Science Foundation (NSF) of China, No.62206156, No.62306074, No.62276155, No.72004127, and No.62206157; in part by the NSF of Shandong Province, No.ZR2024QF104, No.ZR2021MF040 and No.ZR2022QF047; in part by the Key R&D Program of Shandong Province, China (Major Scientific and Technological Innovation Projects), No.2022CXGC020107; in part by the Natural Science Basis Research Plan in Shaanxi Province of China under Grant No.2025JCJCQN-091; in part the Key Research and Development Program of Shaanxi under Grant No.2024GX-YBXM-556

References

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Aklilu, J.; Wang, X.; and Yeung-Levy, S. 2024. Zero-shot Action Localization via the Confidence of Large Vision-Language Models. *arXiv preprint arXiv:2410.14340*.

Bai, S.; Chen, K.; Liu, X.; Wang, J.; Ge, W.; Song, S.; Dang, K.; Wang, P.; Wang, S.; Tang, J.; et al. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Bosetti, M.; Zhang, S.; Liberatori, B.; Zara, G.; Ricci, E.; and Rota, P. 2024. Text-Enhanced Zero-Shot Action Recognition: A Training-Free Approach. In *International Conference on Pattern Recognition*, 327–342. Springer.

Chen, S.; Lan, X.; Yuan, Y.; Jie, Z.; and Ma, L. 2024. Timemarker: A versatile video-llm for long and short video understanding with superior temporal localization ability. *arXiv preprint arXiv:2411.18211*.

Han, C.; Wang, H.; Kuang, J.; Zhang, L.; and Gui, J. 2025. Training-Free Zero-Shot Temporal Action Detection with Vision-Language Models. *arXiv preprint arXiv:2501.13795*.

Heilbron, F.; Escorcia, V.; Ghanem, B.; and Niebles, J. 2019. A largescale video benchmark for human activity understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, 2015. 961*, volume 970.

Idrees, H.; Zamir, A. R.; Jiang, Y.-G.; Gorban, A.; Laptev, I.; Sukthankar, R.; and Shah, M. 2017. The thumos challenge on action recognition for videos "in the wild". *Computer Vision and Image Understanding*, 155: 1–23.

Ju, C.; Han, T.; Zheng, K.; Zhang, Y.; and Xie, W. 2022. Prompting visual-language models for efficient video understanding. In *European Conference on Computer Vision*, 105–124. Springer.

Kim, H.-J.; Hong, J.-H.; Kong, H.; and Lee, S.-W. 2024. Te-tad: Towards full end-to-end temporal action detection via time-aligned coordinate expression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18837–18846.

Kim, J.; Lee, M.; and Heo, J.-P. 2023. Self-feedback detr for temporal action detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 10286–10296.

Li, J.; Li, D.; Savarese, S.; and Hoi, S. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, 19730–19742. PMLR.

Li, Z.; Zhong, Y.; Song, R.; Li, T.; Ma, L.; and Zhang, W. 2024. DeTAL: open-vocabulary temporal action localization with decoupled networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Liberatori, B.; Conti, A.; Rota, P.; Wang, Y.; and Ricci, E. 2024. Test-time zero-shot temporal action localization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 18720–18729.

Lin, B.; Ye, Y.; Zhu, B.; Cui, J.; Ning, M.; Jin, P.; and Yuan, L. 2023. Video-llava: Learning united visual representation by alignment before projection. *arXiv preprint arXiv:2311.10122*.

Liu, A.; Feng, B.; Xue, B.; Wang, B.; Wu, B.; Lu, C.; Zhao, C.; Deng, C.; Zhang, C.; Ruan, C.; et al. 2024a. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Liu, F.; Cheng, Z.; Zhu, L.; Gao, Z.; and Nie, L. 2021. Interest-aware message-passing GCN for recommendation. In *Proceedings of the web conference 2021*, 1296–1305.

Liu, H.; Li, C.; Li, Y.; and Lee, Y. J. 2024b. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 26296–26306.

Liu, S.; Zhang, C.-L.; Zhao, C.; and Ghanem, B. 2024c. End-to-end temporal action detection with 1b parameters across 1000 frames. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 18591–18601.

Maaz, M.; Rasheed, H.; Khan, S.; and Khan, F. S. 2023. Video-chatgpt: Towards detailed video understanding via large vision and language models. *arXiv preprint arXiv:2306.05424*.

Nag, S.; Zhu, X.; Song, Y.-Z.; and Xiang, T. 2022a. Proposal-free temporal action detection via global segmentation mask learning. In *European Conference on Computer Vision*, 645–662. Springer.

Nag, S.; Zhu, X.; Song, Y.-Z.; and Xiang, T. 2022b. Zero-shot temporal action detection via vision-language prompting. In *European conference on computer vision*, 681–697. Springer.

Phan, T.; Vo, K.; Le, D.; Doretto, G.; Adjeroh, D.; and Le, N. 2024. Zeetad: Adapting pretrained vision-language model for zero-shot end-to-end temporal action detection. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 7046–7055.

Qing, Z.; Su, H.; Gan, W.; Wang, D.; Wu, W.; Wang, X.; Qiao, Y.; Yan, J.; Gao, C.; and Sang, N. 2021. Temporal context aggregation network for temporal action proposal refinement. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 485–494.

Shao, J.; Wang, X.; Quan, R.; Zheng, J.; Yang, J.; and Yang, Y. 2023. Action sensitivity learning for temporal action localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 13457–13469.

Shi, D.; Cao, Q.; Zhong, Y.; An, S.; Cheng, J.; Zhu, H.; and Tao, D. 2023a. Temporal action localization with enhanced instant discriminability. *arXiv preprint arXiv:2309.05590*.

Shi, D.; Zhong, Y.; Cao, Q.; Ma, L.; Li, J.; and Tao, D. 2023b. Tridet: Temporal action detection with relative boundary modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18857–18866.

Wang, H.; Xu, Z.; Cheng, Y.; Diao, S.; Zhou, Y.; Cao, Y.; Wang, Q.; Ge, W.; and Huang, L. 2024. Grounded-videollm: Sharpening fine-grained temporal grounding in video large language models. *arXiv preprint arXiv:2410.03290*.

Wang, Y.; Li, K.; Li, Y.; He, Y.; Huang, B.; Zhao, Z.; Zhang, H.; Xu, J.; Liu, Y.; Wang, Z.; et al. 2022. Internvideo: General video foundation models via generative and discriminative learning. *arXiv preprint arXiv:2212.03191*.

Wu, Y.; Hu, X.; Sun, Y.; Zhou, Y.; Zhu, W.; Rao, F.; Schiele, B.; and Yang, X. 2025. Number it: Temporal grounding videos like flipping manga. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 13754–13765.

Xia, K.; Wang, L.; Zhou, S.; Zheng, N.; and Tang, W. 2022. Learning to refactor action and co-occurrence features for temporal action localization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 13884–13893.

Zhan, Y.-W.; Liu, F.; Luo, X.; Xu, X.-S.; Nie, L.; and Kankanhalli, M. 2025. Enhancing HOI Detection with Contextual Cues from Large Vision-Language Models. In *Proceedings of the 33rd ACM International Conference on Multimedia*, 8557–8566.

Zhang, C.-L.; Wu, J.; and Li, Y. 2022. Actionformer: Localizing moments of actions with transformers. In *European Conference on Computer Vision*, 492–510. Springer.

Zhang, H.; Zhang, P.; Hu, X.; Chen, Y.-C.; Li, L.; Dai, X.; Wang, L.; Yuan, L.; Hwang, J.-N.; and Gao, J. 2022. Glipv2: Unifying localization and vision-language understanding. *Advances in Neural Information Processing Systems*, 35: 36067–36080.

Zhao, C.; Thabet, A. K.; and Ghanem, B. 2021. Video self-stitching graph network for temporal action localization. In *Proceedings of the IEEE/CVF international conference on computer vision*, 13658–13667.

Zhao, Y.; Xiong, Y.; Wang, L.; Wu, Z.; Tang, X.; and Lin, D. 2017. Temporal action detection with structured segment networks. In *Proceedings of the IEEE international conference on computer vision*, 2914–2923.

Zhao, Z.; Wang, D.; and Zhao, X. 2023. Movement enhancement toward multi-scale video feature representation for temporal action detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 13555–13564.

Zhu, Y.; Zhang, G.; Tan, J.; Wu, G.; and Wang, L. 2024. Dual detrs for multi-label temporal action detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18559–18569.