TempSamp-R1: Effective Temporal Sampling with Reinforcement Fine-Tuning for Video LLMs

Yunheng Li[1] **Jing Cheng**[2] **Shaoyong Jia**[2] **Hangyi Kuang**[1]
Shaohui Jiao[2] **Qibin Hou**[1,3†] **Ming-Ming Cheng**[1,3]
[1]VCIP, School of Computer Science, Nankai University,
[2]ByteDance Inc., [3]NKIARI, Futian, Shenzhen
yunhengli@mail.nankai.edu.cn

Abstract

This paper introduces TempSamp-R1, a new reinforcement fine-tuning framework designed to improve the effectiveness of adapting multimodal large language models (MLLMs) to video temporal grounding tasks. We reveal that existing reinforcement learning methods, such as Group Relative Policy Optimization (GRPO), rely on on-policy sampling for policy updates. However, in tasks with large temporal search spaces, this strategy becomes both inefficient and limited in performance, as it often fails to identify temporally accurate solutions. To address this limitation, TempSamp-R1 leverages ground-truth annotations as off-policy supervision to provide temporally precise guidance, effectively compensating for the sparsity and misalignment in on-policy solutions. To further stabilize training and reduce variance in reward-based updates, TempSamp-R1 provides a non-linear soft advantage computation method that dynamically reshapes the reward feedback via an asymmetric transformation. By employing a hybrid Chain-of-Thought (CoT) training paradigm, TempSamp-R1 optimizes a single unified model to support both CoT and non-CoT inference modes, enabling efficient handling of queries with varying reasoning complexity. Experimental results demonstrate that TempSamp-R1 outperforms GRPO-based baselines, establishing new state-of-the-art performance on benchmark datasets: Charades-STA (R1@0.7: 52.9%, +**2.7**%), ActivityNet Captions (R1@0.5: 56.0%, +**5.3**%), and QVHighlights (mAP: 30.0%, +**3.0**%). Moreover, TempSamp-R1 shows robust few-shot generalization capabilities under limited data. Code is available at https://github.com/HVision-NKU/TempSamp-R1.

1 Introduction

Multimodal Large Language Models (MLLMs) [6, 21, 30, 31, 34, 40, 51, 58, 64] have demonstrated impressive capabilities in comprehending video content by following general human instructions and effectively interpreting visual content. However, their application to temporal video understanding tasks, such as temporal grounding [1, 22, 25, 35] and highlight detection [24, 43, 62], remains challenging, as these tasks require precise spatio-temporal understanding over long video sequences. A common approach is Supervised Fine-Tuning (SFT) [13, 17, 18, 28, 41, 45, 49, 54, 66], which aligns model predictions with static ground-truth timestamps using deterministic supervision. However, these methods often exhibit limited effectiveness, as models tend to overfit to deterministic timestamp supervision and fail to acquire the temporal reasoning required for flexible event localization [33, 60].

Recent approaches [10, 33, 60, 72] attempt to address these challenges using reinforcement learning (RL) frameworks. In particular, Group Relative Policy Optimization (GRPO) [47] improves temporal

†Corresponding author.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

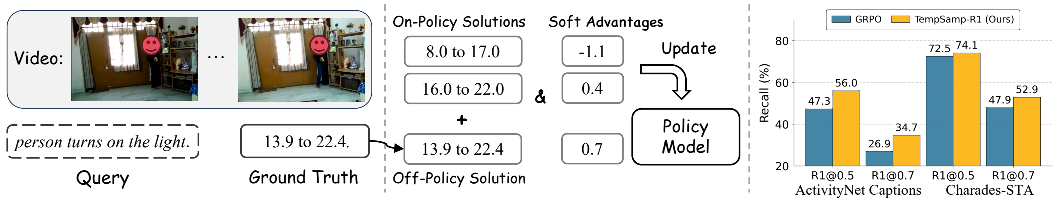


Figure 1: TempSamp-R1 integrates high-quality off-policy solutions with on-policy sampling, combined with soft advantage estimation to enable stable policy updates. It outperforms GRPO, which relies solely on on-policy sampling, on both Charades-STA and ActivityNet Captions.

performance by updating policies via grouped comparisons of sampled solutions, mitigating overfitting to static annotations. GRPO-based methods, such as TimeZero [60] and VideoChat-R1 [33], optimize models via task-specific rewards (e.g., temporal Intersection-over-Union (IoU)) to more effectively align visual dynamics with timestamped semantics in long videos. Despite these advancements, these methods still face a critical limitation: ***The vast temporal search space in temporal grounding tasks severely hinders effective exploration.*** This issue is empirically shown in Fig. 4, when employing GRPO, purely on-policy optimization leads to low and unstable top-1 IoU rewards, particularly on ActivityNet Captions, reflecting unstable early updates, ineffective learning under sparse supervision, and premature convergence to suboptimal solutions.

Notably, most video understanding datasets provide high-quality annotations (e.g., event timestamps) that offer precise supervision for grounding tasks. However, existing GRPO methods treat these annotations solely as evaluation (e.g., computing IoU) rather than dynamic learning sources, leading to suboptimal policy updates. Motivated by this observation, we propose TempSamp-R1, a new reinforcement finetuning framework that integrates on-policy generation with off-policy guidance to facilitate more stable and efficient policy optimization. As illustrated in Fig. 1, TempSamp-R1 incorporates high-quality and instruction-aligned solutions from external sources (e.g., ground truth annotations) as off-policy guidance, providing temporally precise supervision to compensate for the sparsity and misalignment often encountered in on-policy samplings.

However, since off-policy solutions are not sampled from the on-policy model, directly using their rewards can introduce substantial discrepancies in reward distribution, resulting in biased advantage estimation. This estimation bias may suppress high-quality on-policy samplings that diverge from the off-policy solutions, thereby limiting the policy's ability to generalize and explore alternative solutions effectively. To mitigate this, TempSamp-R1 introduces a non-linear soft advantage estimation mechanism inspired by the principles of adaptive reward shaping [15, 39]. To be specific, instead of treating all rewards uniformly, our method distinguishes the learning dynamics between high-reward and low-reward solutions by compressing the advantage values of near-optimal solutions and amplifying the relative reward gaps among suboptimal ones. This asymmetric shaping generates more informative gradients and facilitates stable policy refinement.

By incorporating a hybrid Chain-of-Thought (CoT) [61] training paradigm into a unified model, TempSamp-R1 achieves robust performance under both CoT and non-CoT reasoning modes, which are also shown to be complementary. We evaluate TempSamp-R1 through comprehensive experiments on temporal video understanding benchmarks, including temporal video grounding and video highlight detection. Extensive experiments show that TempSamp-R1 consistently outperforms various SFT-based and GRPO-based methods. Specifically, TempSamp-R1 improves temporal grounding recall accuracy on Charades-STA [11] (R1@0.7: 47.9% $\rightarrow$ 52.9%) and ActivityNet Captions [22] (R1@0.5: 50.7% $\rightarrow$ 56.0%), while enhancing highlight detection in QVHighlights [24] (mAP: 27.0% $\rightarrow$ 30.0%). Notably, TempSamp-R1 maintains competitive performance under limited supervision, highlighting its strong generalization capacity in few-shot scenarios.

2 Related Work

Reinforcement finetuning. Reinforcement learning has emerged as a powerful paradigm for enhancing the reasoning capabilities of large language models and MLLMs [7–9, 12, 19, 38, 52]. For instance, OpenAI's o1 model [19], DeepSeek-R1 [12] and Qwen3 [53] apply RL to generate intermediate reasoning traces before producing final responses, thereby improving performance on complex reasoning tasks. Existing RL fine-tuning approaches can be broadly categorized into reward model-based methods and direct preference optimization techniques. Reward model-based methods, such as RLHF [4], PPO [46], and RLAIF [23], rely on a separately trained reward model to guide policy updates. In contrast, direct preference optimization methods, including DPO [44], IPO [2] and ORPO [16], bypass explicit reward modeling by optimizing preferences directly. GRPO [47] builds upon this paradigm by introducing on-policy sampling and groupwise preference evaluation, enabling dynamic policy refinement from richer comparative samplings. By leveraging comparisons between multiple candidate solutions, GRPO-based methods capture richer alignment supervision than standard SFT, which only learns from a single reference solution [27, 48, 50, 65, 70, 71]. Nonetheless, GRPO's performance still hinges on the diversity and informativeness of samplings, as uninformative comparisons may lead to optimization bias or degraded policy exploration.

Temporal video grounding. Temporal video understanding tasks, such as temporal grounding and highlight detection, require models to accurately identify and describe events within untrimmed video sequences [5, 20, 36, 37, 42, 67, 69, 73]. SFT has been the predominant approach for adapting MLLMs to these tasks [3, 32, 55–57, 74]. Models like TimeChat [45] employ video-text pretraining strategies to align frame-level features with textual descriptions. However, these methods struggle to achieve precise temporal localization due to the limitations in modeling long-range temporal dependencies and the tendency to rely on learned language patterns over visual cues. To address these challenges, RL has been introduced as a fine-tuning strategy to enhance the temporal reasoning capabilities of MLLMs. Recent methods, including TimeZero [60], R1-Omni [72], and VideoChat-R1 [33], utilize GRPO to fine-tune models on spatio-temporal perception tasks. These methods emphasize the design of reward functions to guide the model toward more accurate temporal localization. Our method also builds on the GRPO framework but differs from prior work in that it replaces random on-policy sampling with off-policy solutions to alleviate reward sparsity. To facilitate more efficient and stable policy optimization, we introduce a non-linear soft advantage estimation that dynamically reshapes advantage values to smooth gradient updates.

3 Methodology

To enable effective exploration beyond the limitations of on-policy learning, we propose TempSamp-R1, which combines on-policy generation with off-policy guidance and enhances training stability through soft advantage estimation. As shown in Fig. 2, our method introduces a mixed-policy training strategy based on GRPO, integrating high-quality external solutions (e.g., ground-truth annotations) into the policy optimization process. To stabilize training, we develop a soft advantage estimation mechanism that decouples and shapes reward to reduce gradient variance and adjust advantage bias, thereby promoting robust exploration and convergence.

3.1 Preliminaries

GRPO is a sample-efficient policy optimization algorithm designed to optimize policy models (e.g., MLLMs) by comparing groups of solutions, thereby eliminating the need for an independent value model and reducing computational overhead. Given a query q, GRPO samples a group of G outputs $\{o_1, o_2, \ldots, o_G\}$ from the current policy model π_θ and computes the corresponding rewards $\{r_1, r_2, \ldots, r_G\}$ using a task-specific reward function that evaluates output quality with respect to ground-truth annotations and predefined specific rules (e.g., IoU). The advantage for each solution o_i is then computed as $A_i = \frac{r_i - \mu}{\sigma}$, where μ and σ denote the mean and standard deviation of the group rewards, respectively. This group-normalized advantage serves as the core optimization direction in GRPO, as it adaptively amplifies preferences for outputs that exhibit relatively higher quality within the sampled group. To update the policy, GRPO reuses the same solutions sampled from the previous policy $\pi_{\theta_{\text{old}}}$ and re-evaluates their likelihoods under the current policy π_θ. The update applies importance weighting with a clipping mechanism and incorporates a KL-regularization term to constrain deviation from a reference policy π_{ref}:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \frac{1}{G}\sum_{i=1}^{G}\left[\min\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{\text{old}}}(o_i|q)}A_i, \text{clip}\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{\text{old}}}(o_i|q)}, 1-\epsilon, 1+\epsilon\right)A_i\right) - \beta\,\text{KL}(\pi_\theta||\pi_{\text{ref}})\right], \quad (1)$$

Here, ϵ and β denote the clipping range and the KL-divergence penalty weight, respectively. These components collectively impose constraints on policy updates and mitigate instability during the

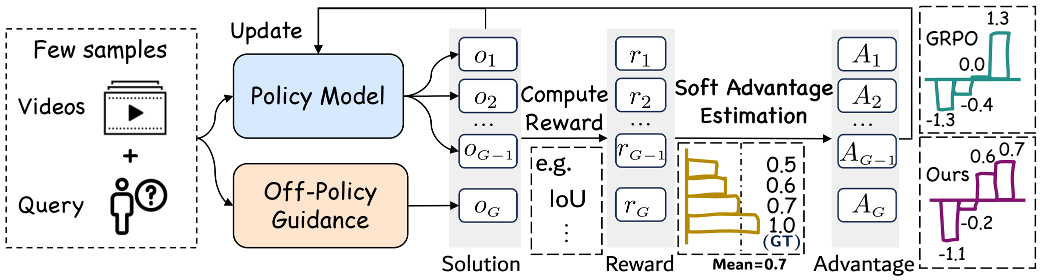


Figure 2: Overview of the TempSamp-R1 framework used to fine-tune the multimodal policy model. Given a few training examples, both the policy model and the off-policy guidance are used to generate solutions. Rewards are computed for each solution, and a soft advantage estimation module transforms raw rewards into standardized advantages for stable policy optimization. Right: Comparison of normalized advantages from GRPO (top) and our method (bottom), illustrating improved advantage discrimination. For clarity, the reference model and KL penalty are omitted.

optimization process. Recent methods [33, 60] often adopt $\pi_{\theta_{\text{old}}} = \pi_\theta$ to balance training data utilization efficiency and computational cost. Under this configuration, the importance weights collapse to unity, eliminating the need for ratio clipping while ensuring stable learning by updating the policy only once per sampling batch.

3.2 TempSamp-R1

Despite their impressive performance on general vision-language tasks, MLLMs often exhibit limited temporal grounding capabilities in video understanding [3, 56]. As a result, training under GRPO with on-policy sampling leads to slow convergence and a constrained performance upper bound, as the policy model encounters significant challenges in generating temporally precise solutions.

Mix-policy sampling. To address the above limitation, we introduce a mixed-policy training strategy that incorporates external off-policy solutions to provide accurate and query-specific temporal grounding. Although such high-quality solutions can be derived from expert policies, we adopt a more direct and empirically effective alternative by utilizing ground-truth annotations as the external policy guidance. To unify supervision across policy sources, we normalize the advantage values using the joint distribution of on-policy and off-policy rewards. Specifically, for each query, we sample $G-1$ solutions from the current policy and include one external off-policy solution (e.g., ground-truth). The normalized advantage for each solution with reward r_i is then computed as:

$$A_i = \frac{r_i - \text{mean}(r_1, r_2, \ldots, r_{G-1} \cup r_G)}{\text{std}(r_1, r_2, \ldots, r_{G-1} \cup r_G)}, \tag{2}$$

where r_G denotes the reward from the external off-policy solution. While the integration of off-policy solutions enhances the diversity and quality of training supervision, it also introduces skewed reward distributions that can destabilize policy optimization. In particular, when the off-policy solution consistently attains the highest reward, even marginal deviations among the remaining rewards (such as low inter-sample variance or tightly clustered suboptimal rewards) can render the normalization process numerically unstable. As illustrated in Fig. 2, the inclusion of off-policy solutions with exceptionally high rewards elevates the intra-group mean reward. In this scenario, the original GRPO algorithm computes negative advantages for all on-policy generated solutions, including those of high quality. This misestimation adversely affects advantage calculation, disrupts gradient updates, and diminishes exploration, ultimately causing premature convergence to suboptimal solutions. To mitigate these adverse effects and ensure stable and reliable advantage estimation, we propose three alternative strategies that regulate the contribution of off-policy supervision.

Reward downscaling. A straightforward mitigation strategy involves explicitly bounding the off-policy reward by scaling it to a fixed fraction (e.g., 80%) of the maximum possible value. This heuristic prevents off-policy solutions from dominating the advantage computation, mitigating distributional shift, and preserving gradient stability. However, the fixed nature of this scaling may suppress valuable learning supervision when off-policy solutions offer genuinely optimal samplings.

Advantage anchoring. To leverage off-policy supervision while mitigating distributional bias, we introduce an anchoring mechanism that decouples external solutions from on-policy advantage estimation. Specifically, off-policy samples are excluded from the computation of group statistics and do not participate in normalization. Instead, their advantage values are computed by scaling the maximum on-policy advantage within the group:

$$A_G = \lambda_{\text{off}} \cdot \max\left\{A_i \mid i \in \{1, 2, \cdots, G-1\}\right\}, \tag{3}$$

where $\lambda_{\text{off}} = 1.2$ is a fixed scaling factor. This anchoring preserves the supervision from off-policy data while constraining its influence, maintaining stability and consistency in policy gradient updates.

Non-linear reward shaping. To improve stability under skewed reward distributions, we apply a non-linear transformation to the rewards prior to advantage computation. This transformation is defined as an asymmetric piecewise function that compresses high rewards and expands low rewards. The shaping reward $\tilde{r}_i$ is defined as:

$$\tilde{r}_i = \begin{cases} \tau + \alpha_1 \cdot \ln\left((r_i - \tau) + 1\right), & r_i \geq \tau \\ \tau - \dfrac{e^{\alpha_2 \cdot (\tau - r_i)} - 1}{e^{\alpha_2} - 1}. & r_i < \tau \end{cases} \tag{4}$$

Here, $\tau = 0.8$ is the reward threshold, $\alpha_1 = 0.01$ controls compression above the threshold, and $\alpha_2 = 1$ governs expansion below it. The logarithmic branch mitigates gradient spikes from optimal solutions, while the exponential branch increases contrast among suboptimal samplings.

Each of the above strategies provides an alternative mechanism to mitigate instability caused by incorporating strong off-policy solutions. By independently adjusting rewards, advantage scales, or reward distributions, these methods offer flexible design choices to control the influence of external off-policy and improve the stability of policy optimization in multimodal video understanding.

3.3 Training

We adopt a two-phase training scheme with task-specific reward functions to support diverse video understanding tasks. In the initialization phase, the model is optimized to generate accurate final answers without explicit reasoning. Building on this, we incorporate format rewards to encourage the generation of intermediate reasoning steps alongside final outputs. To operationalize this strategy across different tasks, we define a suite of task-specific reward functions that directly guide policy optimization toward task-relevant behaviors.

IoU reward. For temporal localization tasks, the model is required to predict an event interval $[t_{\text{p}}^{\text{s}}, t_{\text{p}}^{\text{e}}]$ conditioned on a given query. To quantify prediction accuracy, we define a reward rule based on the IoU with the ground truth interval $[t_{\text{g}}^{\text{s}}, t_{\text{g}}^{\text{e}}]$, computed as $R_{\text{IoU}} = (\min(t_p^e, t_g^e) - \max(t_p^s, t_g^s))/(\max(t_p^e, t_g^e) - \min(t_p^s, t_g^s))$. The R_{IoU} is computed as the ratio between the intersection and the union of the two intervals, serving as a direct measure of temporal alignment accuracy.

Timestamp matching reward. For highlight detection tasks, the model jointly predicts temporal boundaries and associated saliency scores. To evaluate both the structural and semantic quality of these predictions, we define a composite reward: $R_{\text{ts}} = \lambda_{\text{rec}} \cdot \text{F2} + \lambda_{\text{score}} \cdot \frac{1}{1+\text{WMSE}}$. Here, the F2 score measures the temporal alignment by computing a recall-weighted F-measure over matched timestamps, emphasizing recall to better capture relevant highlights. The Weighted Mean Squared Error (WMSE) assesses the fidelity of predicted saliency scores, with weights derived from the squared ground-truth scores to emphasize high-saliency regions. We set $\lambda_{\text{rec}} = 0.6$ and $\lambda_{\text{score}} = 0.4$ to prioritize semantic fidelity in salient regions while ensuring temporal coverage.

Format reward. To promote structured output in reasoning tasks, we introduce a format reward that enforces conformity to a predefined schema. The model is expected to generate reasoning enclosed in <Think>...</Think> and final answers in <Answer>...</Answer>. The reward is set to 1 if the output matches the required structure based on regular expression validation, and 0 otherwise.

4 Experiments

Implementation details. Our experiments are conducted using the Qwen2.5-VL-7B-Instruct model [3]. To ensure a fair comparison with prior efficient video fine-tuning methods [33, 60],

Method	Type	Charades-STA mIoU	R1@0.3	R1@0.5	R1@0.7	ActivityNet Captions mIoU	R1@0.3	R1@0.5	R1@0.7	QVHighlights mAP	HIT@1
Supervised Fine-Tuning (SFT) Methods											
UnLoc-L [63]	SFT	-	-	60.8	38.4	-	-	48.3	30.2	-	-
Timechat [45]	SFT	-	-	46.7	23.7	-	-	-	-	21.7	37.9
HawkEye [59]	SFT	49.3	72.5	58.3	28.8	39.1	55.9	34.7	17.9	-	-
TRACE [14]	SFT	-	-	61.7	41.4	-	-	37.7	24.0	-	-
VideoChat-T [68]	SFT	-	79.4	67.1	43.0	-	-	-	-	27.0	<u>55.3</u>
iMOVE [29]	SFT	57.9	79.8	68.5	45.3	49.3	67.2	50.7	32.4	-	-
Reinforcement Learning (RL) Methods based on Qwen2.5-VL-7B											
Qwen2.5-VL-7B [3]*	-	49.7	73.4	54.4	30.3	33.1	45.2	29.7	18.1	19.7	34.1
VideoChat-R1 [33]	RL	60.8	-	71.7	50.2	-	-	-	-	-	-
VideoChat-R1-thinking [33]	RL	59.9	-	70.6	47.2	-	-	-	-	-	-
TimeZero [60]	RL	-	<u>83.3</u>	72.5	47.9	-	68.6	47.3	26.9	-	-
TempSamp-R1(no-CoT)	RL	<u>61.7</u>	<u>83.3</u>	<u>73.6</u>	<u>52.2</u>	<u>52.1</u>	<u>72.8</u>	<u>55.4</u>	<u>34.2</u>	**30.0**	**57.6**
TempSamp-R1 (CoT)	RL	**62.1**	**83.6**	**74.1**	**52.9**	**52.4**	**73.4**	**56.0**	**34.7**	<u>28.3</u>	54.9
TempSamp-R1 Mixed CoT	RL	64.2	85.0	76.0	56.3	54.9	75.7	58.7	37.6	29.3	63.7

Table 1: Performance comparison of Charades-STA, ActivityNet Captions, and QVHighlights datasets. Our TempSamp-R1 supports both CoT and no-CoT reasoning within a single unified model, and achieves strong performance across all datasets. The TempSamp-R1 Mixed CoT selects the better prediction between CoT and no-CoT for each query.

Method	mIoU	R1@0.3	R1@0.5	R1@0.7
SFT	20.6	30.2	16.7	7.9
GRPO	30.7	45.0	27.5	12.9
TempSamp-R1	**34.7**	**50.9**	**32.2**	**16.2**

Table 2: Out-of-domain generalization performance from Charades-STA to ActivityNet.

we standardize the input preprocessing pipeline. Specifically, all videos are temporally downsampled to 2 frames per second (FPS) and resized to approximately 2.8 million pixels per frame. Training is performed on four NVIDIA A100 GPUs with a batch size of 1 per GPU. For GRPO-based training, each question is associated with a total of 4 solutions, consisting of $G{=}3$ on-policy samplings and 1 off-policy solution. This setting balances computational efficiency and diversity in policy learning. For Charades-STA (Tab. 1), we increase the number of solutions to 8 to better accommodate the task's higher compositional complexity.

Benchmarks. We evaluate our model on the temporal grounding task using the Charades-STA, ActivityNet Captions, and QVHighlights datasets. Following established practices [45, 68], we report Recall@1 (R1@) at Intersection over Union (IoU) thresholds of 0.3, 0.5, and 0.7. Additionally, we compute the mean IoU (mIoU) across all test samples to evaluate overall localization accuracy. The QVHighlights dataset evaluates using mean Average Precision (mAP) at IoU thresholds of 0.5 and 0.75, and HIT@1, which indicates whether the top-ranked clip is labeled as "Very Good."

4.1 Main results

Fine-tuning performance. We evaluate the effectiveness of our proposed method, TempSamp-R1, across three standard benchmarks for temporal grounding: Charades-STA, ActivityNet Captions, and QVHighlights. Tab. 1 compares our approach with a wide range of baselines, including zero-shot, supervised fine-tuning (SFT) methods, and reinforcement learning (RL) based approaches. Compared to state-of-the-art SFT baselines, TempSamp-R1 achieves stronger performance. On Charades-STA, it obtains 74.1% R1@0.5 and 52.9% R1@0.7, outperforming iMOVE by +5.6% and +7.6% respectively. In comparison to existing RL-based approaches, TempSamp-R1 achieves consistent gains across all benchmarks. For instance, it surpasses VideoChat-R1 by +2.4% R1@0.5 and TimeZero by +5.0% R1@0.7 on Charades-STA. On ActivityNet Captions, it exceeds TimeZero by +8.7% R1@0.5 and

*For Qwen2.5-VL-7B, the results on Charades-STA and ActivityNet Captions are reproduced from [26], whereas the results on QVHighlights are obtained from our implementation.

Method	50 videos		100 videos		200 videos		500 videos		
	R1@0.5	mIoU	R1@0.5	mIoU	R1@0.5	mIoU	R1@0.5	mIoU	Training Time
SFT	44.8	41.9	46.5	42.6	45.2	42.7	51.4	46.2	93 min
GRPO	36.2	38.4	39.3	40.8	43.5	43.8	55.3	49.8	338 min
TempSamp-R1 (Ours)	**46.7**	**44.7**	**54.0**	**49.1**	**58.2**	**51.8**	**64.0**	**55.1**	218 min

Table 3: Few-shot performance comparison of SFT, GRPO, and our proposed method on Charades-STA under varying training sample sizes (50, 100, 200, 500). All models were trained for 3 epochs.

Method	R1@0.3	R1@0.5	R1@0.7
GRPO (baseline)	81.2	68.9	46.0
Mixed-policy	77.8	63.0	41.3
Reward downscaling	81.2	70.3	48.1
Advantage anchoring	81.8	70.7	49.1
Non-linear reward shaping	**82.9**	**72.1**	**49.6**

Table 4: Ablation results comparing GRPO with enhanced variants incorporating mixed-policy rewards and alternative advantage shaping strategies.

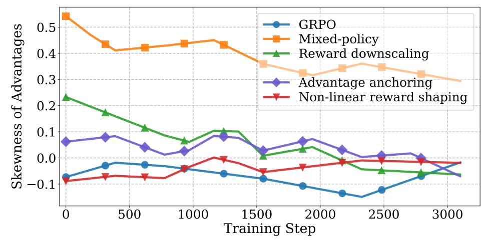


Figure 3: Skewness of the advantage distributions during training for different variants.

+7.8% R1@0.7. We observe that CoT prompting at inference consistently improves performance on Charades-STA and ActivityNet Captions, indicating that explicit reasoning is beneficial for tasks involving complex temporal dependencies. In contrast, on QVHighlights, the TempSamp-R1 with no-CoT performs better, indicating that direct prediction is more suitable for highlight detection, where explicit reasoning may be redundant or distracting. We further explore a mixed variant, TempSamp-R1 Mixed CoT, which selects the better output between CoT and no-CoT predictions for each query. This strategy consistently outperforms either individual reasoning mode, underscoring their complementary roles. Representative examples in Fig. 6 illustrate how each reasoning mode excels under different semantic and temporal conditions.

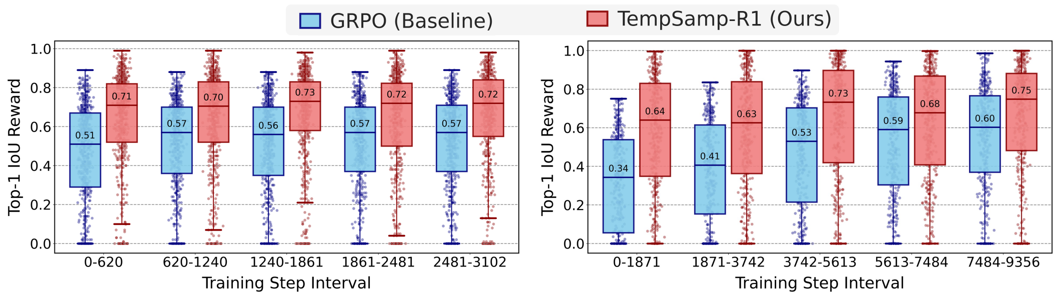
Out-of-domain generalization. To assess the cross-dataset transferability of different approaches, we conduct out-of-domain evaluations where all models are trained on Charades-STA and directly tested on ActivityNet Captions. As shown in Tab. 2, TempSamp-R1 consistently outperforms both SFT and GRPO across all metrics on both datasets. On ActivityNet Captions, it achieves improvements of +4.0% mIoU and +4.7% R1@0.5 over GRPO. These results suggest that off-policy supervision and soft advantage shaping jointly enhance the cross-domain transferability of TempSamp-R1.

Few-shot performance. We evaluate our method under few-shot settings on the Charades-STA dataset, training with 50, 100, 200, and 500 videos, each for 3 epochs. Table 3 presents the performance comparison among SFT, GRPO, and our proposed method. Our method consistently outperforms both SFT and GRPO across all training sizes. Notably, with just 50 training samples, our method achieves a mIoU of 44.7%, surpassing SFT by +2.8%. As the number of training samples increases, the performance gap widens. With 500 samples, our method attains an R1@0.5 of 64.0%, outperforming SFT by +12.6%, and GRPO by +8.7%, respectively. In terms of training efficiency, our method requires 218 minutes for training with 500 samples, which less than GRPO's 338 minutes. These results demonstrate that our method maintains efficient training, highlighting its practicality for real-world applications, where annotated data is limited.

4.2 Component-wise analysis of TempSamp-R1

We analyze our method on Charades-STA and ActivityNet Captions to assess the impact of key components in TempSamp-R1, including off-policy guidance and advantage shaping. Results show that these mechanisms jointly contribute to more stable training and better policy optimization.

Analysis of advantage shaping strategies. We analyze the advantage shaping strategies introduced in Sec. 3.2 through systematic ablations within the TempSamp-R1 framework here. Tab. 4 compares the GRPO baseline against four variants: mixed-policy supervision, reward downscaling, advantage anchoring, and non-linear reward shaping. Directly injecting ground-truth rewards (mixed-policy)



(a) Training on Charades-STA

(b) Training on ActivityNet Caption

Figure 4: Distribution of top-1 IoU rewards under GRPO and TempSamp-R1 on Charades-STA and ActivityNet Captions. TempSamp-R1 exhibits higher median rewards and reduced variance, indicating more stable and effective policy learning.

	IoU=0.3		IoU=0.5		IoU=0.7	
Samplings	GRPO	TempSamp-R1 (+Δ)	GRPO	TempSamp-R1 (+Δ)	GRPO	TempSamp-R1 (+Δ)
2	77.8	80.8 (+3.0)	60.2	67.3 (+7.1)	34.4	44.6 (+10.2)
4	81.0	82.5 (+1.5)	68.9	71.6 (+2.7)	46.0	49.6 (+3.6)
6	81.3	82.6 (+1.3)	69.2	71.7 (+2.5)	49.2	50.3 (+1.1)
8	81.2	82.6 (+1.4)	70.5	71.9 (+1.4)	47.1	50.8 (+3.7)

Table 5: Ablation study on the number of solutions. Our method consistently outperforms the baseline GRPO across all configurations, particularly under higher IoU thresholds.

leads to degraded performance (e.g., 63.0 R1@0.5), which can be attributed to distributional shift and reduced on-policy sampling diversity. Reward downscaling and advantage anchoring partially mitigate this issue, improving R1@0.5 to 70.3 and 70.7, respectively. Non-linear reward shaping achieves the best results. To better understand these results, Fig. 3 analyzes the skewness of the advantage distributions throughout training. GRPO exhibits persistent negative skewness, indicating dominance of low-reward solutions and weak gradient magnitude. In contrast, mixed-policy supervision results in high positive skewness, reflecting over-reliance on a small set of high-reward solutions and poor policy generalization. Our proposed three shaping strategies mitigate these imbalances to varying degrees. Notably, non-linear reward shaping maintains near-zero skewness throughout training, promoting stable optimization and improved grounding performance.

Analysis of reward distribution. To better understand the learning dynamics of different methods, we analyze the distribution of top-1 IoU rewards collected during training. Fig. 4 presents box plots comparing GRPO and methodname on Charades-STA and ActivityNet Captions. GRPO baseline exhibits low median rewards with high variance, especially on ActivityNet Captions, reflecting unstable exploration and frequent convergence to suboptimal solutions. In contrast, methodname yields significantly higher median rewards and notably reduced dispersion. This indicates that the integration of off-policy supervision and non-linear advantage shaping not only improves reward magnitude but also enhances training stability. The compact distribution further suggests that methodname can consistently identify high-quality solutions across different video samples, even in the presence of long temporal sequences and ambiguous queries.

4.3 Ablation study

We conduct ablations to evaluate the individual contributions of sampling strategy and CoT supervision. Results on Charades-STA show that TempSamp-R1 remains effective under limited on-policy samplings, and CoT supervision enhances temporal localization.

Impact of the number of solutions. We conduct an ablation study to evaluate how varying the number of solutions per query affects temporal grounding performance. Specifically, we assessed models with 2, 4, 6, and 8 solutions on the Charades-STA dataset. As shown in Tab. 5, TempSamp-R1 outperforms the baseline GRPO across all configurations. Notably, with only 2 solutions, TempSamp-R1 achieves a remarkable 10.2% absolute improvement in R1@0.7 over GRPO, demonstrating its robustness

Method	Epochs	Training Prompt Think	Training Prompt Answer	Test Prompt Think	Test Prompt Answer	mIoU	R1@0.3	R1@0.5	R1@0.7
Qwen2.5-VL-7B	-	-	-		✓	29.0	44.7	24.2	11.1
(baseline)	-	-	-	✓	✓	28.1	41.8	23.4	11.1
+ TempSamp-R1	1		✓		✓	61.1	82.6	71.9	50.8
	1		✓	✓	✓	54.2	75.9	63.6	40.5
	2		✓		✓	61.5	82.6	73.2	51.2
	2		✓	✓	✓	61.2	82.4	72.8	50.6
	2	✓	✓		✓	61.8	83.3	73.6	52.2
	2	✓	✓	✓	✓	62.1	83.6	74.1	52.9

Table 6: Ablation study on the effects of `<Think>` and `<Answer>` prompts during training and testing under various configurations of our method on the Charades-STA dataset.

τ	R1@0.3	R1@0.5	R1@0.7
0.7	81.6	70.5	48.4
0.8	**82.9**	**72.1**	**49.6**
0.9	81.1	69.4	47.5

(a) Ablation on τ.

α_1	R1@0.3	R1@0.5	R1@0.7
0.05	81.6	70.5	48.4
0.01	**82.9**	**72.1**	**49.6**
0.02	81.9	71.0	49.1

(b) Ablation on α_1.

α_2	R1@0.3	R1@0.5	R1@0.7
0.5	82.2	71.6	49.0
1.0	**82.9**	**72.1**	**49.6**
2.0	81.1	69.5	46.9

(c) Ablation on α_2.

Figure 5: Ablation study on the effects of different hyperparameters τ, α_1, and α_2 in the non-linear reward shaping method.

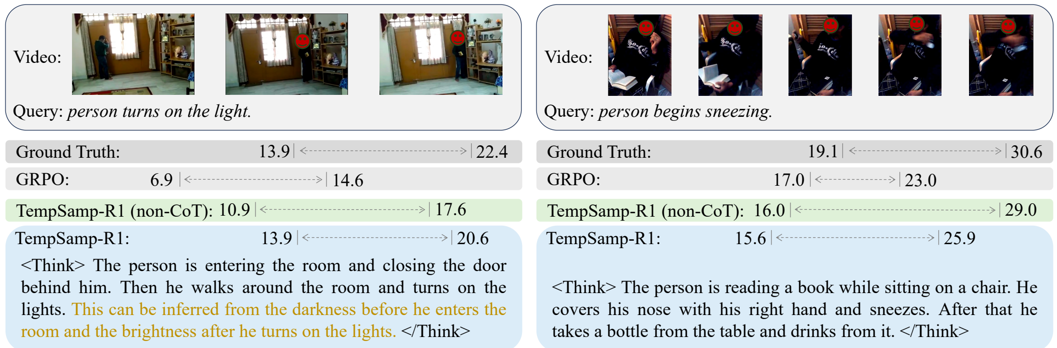
under limited on-policy sampling. While increasing the number of solutions generally leads to improved performance for both methods, the gains for GRPO diminish beyond 4 solutions, whereas TempSamp-R1 continues to benefit from additional on-policy samplings.

Impact of CoT supervision and inference. We perform an ablation study to assess the impact of incorporating `<Think>` and `<Answer>` prompts during training and testing on the Charades-STA dataset. Tab. 6 presents the results across different combinations of training-phase supervision and inference-time prompting. The best performance is achieved when utilizing both `<Think>` and `<Answer>` prompts during training and inference. This suggests that the combination of CoT prompts enhances the model's temporal grounding capabilities. In contrast, applying `<Think>` prompts only during inference, without corresponding supervision during training, results in a substantial performance drop, suggesting sensitivity to mismatched prompting conditions. In addition, training solely with `<Answer>` prompts for two epochs results in reasonable performance but does not match the effectiveness of the hybrid CoT strategy, underscoring the benefits of incorporating CoT prompts during training. Notably, Tab. 6 (last rows) indicate that training with one non-CoT epoch followed by one CoT epoch yields stable performance regardless of test-time prompt design, indicating the model's robustness to CoT prompt variations after hybrid CoT training.

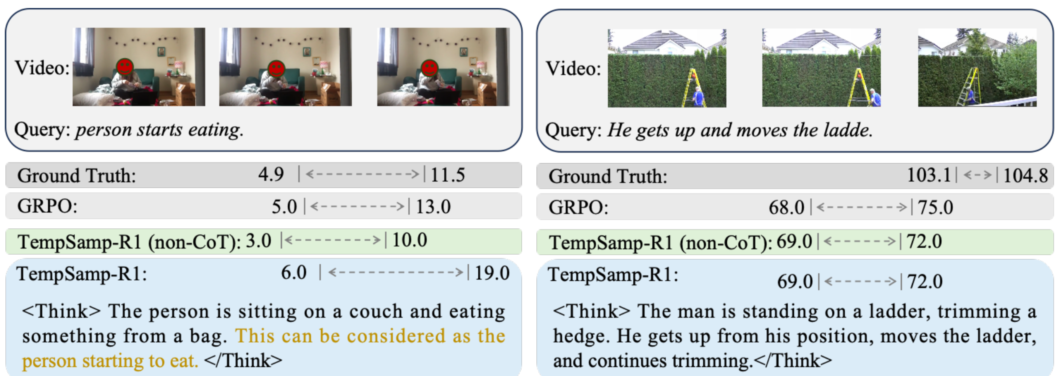
Impact of non-linear reward shaping parameters. We analyze how each parameter in the non-linear reward shaping method influences learning dynamics, as shown in Tab. 5. The threshold τ defines the boundary between high- and low-reward regions. Moderate values maintain balanced optimization, whereas larger τ exacerbate reward imbalance by concentrating rewards on a few high-value solutions, which reduces the model's ability to learn from informative suboptimal solutions. The coefficient α_1 controls logarithmic scaling above the threshold. Smaller values retain gradient diversity, while larger values overly smooth the reward distribution. In contrast, α_2 governs exponential scaling below the threshold. It enhances contrast among lower rewards but can cause instability when set too high. Overall, these findings indicate that the proposed non-linear shaping method is stable within a broad parameter range, with $\tau = 0.8$, $\alpha_1 = 0.01$, and $\alpha_2 = 1.0$ yielding the trade-off between reward contrast and optimization stability.

4.4 Qualitative results

To further illustrate the effectiveness of our method, we present qualitative examples from the Charades-STA and ActivityNet Captions datasets, as shown in Fig. 6. Compared to the GRPO baseline, our model achieves more accurate temporal localization and generates more coherent event reasoning, particularly in challenging or ambiguous cases. A key strength of TempSamp-R1 is its



(a) Qualitative examples where TempSamp-R1 successfully queries. CoT inference shows stronger contextual reasoning for complex queries, while non-CoT mode yields sharper boundaries for straightforward actions.



(b) Failure cases mainly arise from ambiguous annotations and subtle cues that challenge accurate localization.

Figure 6: Qualitative comparisons on temporal grounding using TempSamp-R1. TempSamp denotes evaluation without CoT, while TempSamp-R1 uses the <Think> prompt.

ability to optionally include the <Think> prompt at inference time. In cases requiring reasoning, such as detecting transitions based on subtle visual cues, the CoT prompt improves boundary precision. For instance, when identifying the moment a person turns on a light, the model correctly associates the event with a change in brightness. In contrast, for queries with clear visual markers, the model performs well without reasoning prompts, demonstrating adaptability to different reasoning demands.

However, as shown in the failure cases, errors often stem from ambiguous or narrowly defined ground-truth annotations, which introduce uncertainty in the evaluation. Additionally, some complex queries involve subtle or overlapping visual cues that challenge precise localization, suggesting potential areas for further improvement in both annotation quality and model reasoning.

5 Conclusions

This paper introduces TempSamp-R1, a reinforcement learning framework designed to enhance temporal video understanding in multimodal large language models. TempSamp-R1 integrates off-policy supervision with a non-linear soft advantage mechanism to address the challenges of sparse rewards and unstable policy updates inherent in large temporal search spaces. By promoting effective policy updates, TempSamp-R1 enables more accurate temporal localization of target events described in the input queries. To improve reasoning robustness, it further incorporates a hybrid Chain-of-Thought (CoT) training paradigm, enabling a unified model to perform effectively under both CoT and non-CoT inference modes. Extensive evaluations on benchmarks such as Charades-STA, ActivityNet Captions, and QVHighlights demonstrate that TempSamp-R1 consistently outperforms both SFT-based and existing GRPO-based reinforcement learning methods.

Acknowledgment. This work was supported in part by NSFC (No. 62495061 and No. 62225604) and the Shenzhen Science and Technology Program under Grant No. JCYJ20240813114237048.

References

[1] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *CVPR*, pages 5803–5812, 2017. 1

[2] Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Remi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. A general theoretical paradigm to understand learning from human preferences. In *ICML*, pages 4447–4455. PMLR, 2024. 3

[3] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 3, 4, 5, 6, 14

[4] Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022. 3

[5] 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. pages 9226–9236. IEEE, 2025. 3

[6] Sihan Chen, Handong Li, Qunbo Wang, Zijia Zhao, Mingzhen Sun, Xinxin Zhu, and Jing Liu. Vast: A vision-audio-subtitle-text omni-modality foundation model and dataset. In *NeurIPS*, volume 36, pages 72842–72866, 2023. 1

[7] Huilin Deng, Ding Zou, Rui Ma, Hongchen Luo, Yang Cao, and Yu Kang. Boosting the generalization and reasoning of vision language models with curriculum reinforcement learning. *arXiv preprint arXiv:2503.07065*, 2025. 2

[8] Yihe Deng, Hritik Bansal, Fan Yin, Nanyun Peng, Wei Wang, and Kai-Wei Chang. Openvlthinker: An early exploration to complex vision-language reasoning via iterative self-improvement. *arXiv preprint arXiv:2503.17352*, 2025.

[9] Yifan Du, Zikang Liu, Yifan Li, Wayne Xin Zhao, Yuqi Huo, Bingning Wang, Weipeng Chen, Zheng Liu, Zhongyuan Wang, and Ji-Rong Wen. Virgo: A preliminary exploration on reproducing o1-like mllm. *arXiv preprint arXiv:2501.01904*, 2025. 2

[10] Kaituo Feng, Kaixiong Gong, Bohao Li, Zonghao Guo, Yibing Wang, Tianshuo Peng, Benyou Wang, and Xiangyu Yue. Video-r1: Reinforcing video reasoning in mllms. *arXiv preprint arXiv:2503.21776*, 2025. 1

[11] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. Tall: Temporal activity localization via language query. In *ICCV*, Oct 2017. 2, 15

[12] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025. 2

[13] Yongxin Guo, Jingyu Liu, Mingda Li, Dingxin Cheng, Xiaoying Tang, Dianbo Sui, Qingbin Liu, Xi Chen, and Kevin Zhao. Vtg-llm: Integrating timestamp knowledge into video llms for enhanced video temporal grounding. In *AAAI*, volume 39, pages 3302–3310, 2025. 1

[14] Yongxin Guo, Jingyu Liu, Mingda Li, Qingbin Liu, Xi Chen, and Xiaoying Tang. Trace: Temporal grounding video llm via causal event modeling. *arXiv preprint arXiv:2410.05643*, 2024. 6

[15] Abhishek Gupta, Aldo Pacchiano, Yuexiang Zhai, Sham Kakade, and Sergey Levine. Unpacking reward shaping: Understanding the benefits of reward engineering on sample complexity. In *NeurIPS*, volume 35, pages 15281–15295, 2022. 2

[16] Jiwoo Hong, Noah Lee, and James Thorne. Orpo: Monolithic preference optimization without reference model. *arXiv preprint arXiv:2403.07691*, 2024. 3

[17] Bin Huang, Xin Wang, Hong Chen, Zihan Song, and Wenwu Zhu. Vtimellm: Empower llm to grasp video moments. In *CVPR*, pages 14271–14280, 2024. 1

[18] Bin Huang, Xin Wang, Hong Chen, Zihan Song, and Wenwu Zhu. Vtimellm: Empower llm to grasp video moments. In *CVPR*, pages 14271–14280, June 2024. 1

[19] Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024. 2

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

[21] Peng Jin, Ryuichi Takanobu, Wancai Zhang, Xiaochun Cao, and Li Yuan. Chat-univi: Unified visual representation empowers large language models with image and video understanding. In *CVPR*, pages 13700–13710, June 2024. 1

[22] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *ICCV*, Oct 2017. 1, 2, 15

[23] Harrison Lee, Samrat Phatale, Hassan Mansoor, Kellie Ren Lu, Thomas Mesnard, Johan Ferret, Colton Bishop, Ethan Hall, Victor Carbune, and Abhinav Rastogi. Rlaif: Scaling reinforcement learning from human feedback with ai feedback. 2023. 3

[24] Jie Lei, Tamara L Berg, and Mohit Bansal. Detecting moments and highlights in videos via natural language queries. In *NeurIPS*, volume 34, pages 11846–11858, 2021. 1, 2, 15

[25] Jie Lei, Licheng Yu, Tamara L Berg, and Mohit Bansal. Tvqa+: Spatio-temporal grounding for video question answering. *arXiv preprint arXiv:1904.11574*, 2019. 1

[26] Bo Li, Peiyuan Zhang, Kaichen Zhang, Fanyi Pu, Xinrun Du, Yuhao Dong, Haotian Liu, Yuanhan Zhang, Ge Zhang, Chunyuan Li, and Ziwei Liu. Lmms-eval: Accelerating the development of large multimoal models, March 2024. 6

[27] Cheng Li, Jiexiong Liu, Yixuan Chen, and Yanqin Jia. Video-vot-r1: An efficient video inference model integrating image packing and aoe architecture. *arXiv preprint arXiv:2503.15807*, 2025. 3

[28] Hongyu Li, Jinyu Chen, Ziyu Wei, Shaofei Huang, Tianrui Hui, Jialin Gao, Xiaoming Wei, and Si Liu. Llava-st: A multimodal large language model for fine-grained spatial-temporal understanding, 2025. 1

[29] Jiaze Li, Yaya Shi, Zongyang Ma, Haoran Xu, Feng Cheng, Huihui Xiao, Ruiwen Kang, Fan Yang, Tingting Gao, and Di Zhang. imove: Instance-motion-aware video understanding. *arXiv preprint arXiv:2502.11594*, 2025. 6

[30] 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. 1

[31] Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, Limin Wang, and Yu Qiao. Mvbench: A comprehensive multi-modal video understanding benchmark. In *CVPR*, pages 22195–22206, June 2024. 1

[32] Xinhao Li, Yi Wang, Jiashuo Yu, Xiangyu Zeng, Yuhan Zhu, Haian Huang, Jianfei Gao, Kunchang Li, Yinan He, Chenting Wang, et al. Videochat-flash: Hierarchical compression for long-context video modeling. *arXiv preprint arXiv:2501.00574*, 2024. 3

[33] Xinhao Li, Ziang Yan, Desen Meng, Lu Dong, Xiangyu Zeng, Yinan He, Yali Wang, Yu Qiao, Yi Wang, and Limin Wang. Videochat-r1: Enhancing spatio-temporal perception via reinforcement fine-tuning. *arXiv preprint arXiv:2504.06958*, 2025. 1, 2, 3, 4, 5, 6

[34] 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. 1

[35] Kevin Qinghong Lin, Pengchuan Zhang, Joya Chen, Shraman Pramanick, Difei Gao, Alex Jinpeng Wang, Rui Yan, and Mike Zheng Shou. Univtg: Towards unified video-language temporal grounding. In *ICCV*, pages 2794–2804, 2023. 1

[36] Daizong Liu, Xiaoye Qu, and Wei Hu. Reducing the vision and language bias for temporal sentence grounding. In *ACM MM*, pages 4092–4101, 2022. 3

[37] Shenglan Liu, Aibin Zhang, Yunheng Li, Jian Zhou, Li Xu, Zhuben Dong, and Renhao Zhang. Temporal segmentation of fine-gained semantic action: A motion-centered figure skating dataset. In *AAAI*, volume 35, pages 2163–2171, 2021. 3

[38] Yuqi Liu, Bohao Peng, Zhisheng Zhong, Zihao Yue, Fanbin Lu, Bei Yu, and Jiaya Jia. Seg-zero: Reasoning-chain guided segmentation via cognitive reinforcement. *arXiv preprint arXiv:2503.06520*, 2025. 2

[39] Haozhe Ma, Zhengding Luo, Thanh Vinh Vo, Kuankuan Sima, and Tze-Yun Leong. Highly efficient self-adaptive reward shaping for reinforcement learning. In *ICLR*, 2025. 2

[40] 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. 1

[41] 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

[42] Fangzhou Mu, Sicheng Mo, and Yin Li. Snag: Scalable and accurate video grounding. In *CVPR*, pages 18930–18940, 2024. 3

[43] Long Qian, Juncheng Li, Yu Wu, Yaobo Ye, Hao Fei, Tat-Seng Chua, Yueting Zhuang, and Siliang Tang. Momentor: advancing video large language model with fine-grained temporal reasoning. In *ICML*, pages 41340–41356, 2024. 1

[44] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *NeurIPS*, volume 36, pages 53728–53741, 2023. 3

[45] Shuhuai Ren, Linli Yao, Shicheng Li, Xu Sun, and Lu Hou. Timechat: A time-sensitive multimodal large language model for long video understanding. In *CVPR*, pages 14313–14323, June 2024. 1, 3, 6

[46] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 3

[47] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. 1, 3

[48] Haozhan Shen, Peng Liu, Jingcheng Li, Chunxin Fang, Yibo Ma, Jiajia Liao, Qiaoli Shen, Zilun Zhang, Kangjia Zhao, Qianqian Zhang, et al. Vlm-r1: A stable and generalizable r1-style large vision-language model. *arXiv preprint arXiv:2504.07615*, 2025. 3

[49] Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, et al. Moviechat: From dense token to sparse memory for long video understanding. In *CVPR*, pages 18221–18232, 2024. 1

[50] Huajie Tan, Yuheng Ji, Xiaoshuai Hao, Minglan Lin, Pengwei Wang, Zhongyuan Wang, and Shanghang Zhang. Reason-rft: Reinforcement fine-tuning for visual reasoning. *arXiv preprint arXiv:2503.20752*, 2025. 3

[51] Yunlong Tang, Jing Bi, Siting Xu, Luchuan Song, Susan Liang, Teng Wang, Daoan Zhang, Jie An, Jingyang Lin, Rongyi Zhu, et al. Video understanding with large language models: A survey. *IEEE TCSVT*, 2025. 1

[52] Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, et al. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*, 2025. 2

[53] Qwen Team. Qwen3, April 2025. 2

[54] 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

[55] Yi Wang, Yinan He, Yizhuo Li, Kunchang Li, Jiashuo Yu, Xin Ma, Xinhao Li, Guo Chen, Xinyuan Chen, Yaohui Wang, et al. Internvid: A large-scale video-text dataset for multimodal understanding and generation. *arXiv preprint arXiv:2307.06942*, 2023. 3

[56] Yi Wang, Kunchang Li, Xinhao Li, Jiashuo Yu, Yinan He, Guo Chen, Baoqi Pei, Rongkun Zheng, Zun Wang, Yansong Shi, et al. Internvideo2: Scaling foundation models for multimodal video understanding. In *ECCV*, pages 396–416. Springer, 2024. 4

[57] Yi Wang, Kunchang Li, Yizhuo Li, Yinan He, Bingkun Huang, Zhiyu Zhao, Hongjie Zhang, Jilan Xu, Yi Liu, Zun Wang, et al. Internvideo: General video foundation models via generative and discriminative learning. *arXiv preprint arXiv:2212.03191*, 2022. 3

[58] Yi Wang, Xinhao Li, Ziang Yan, Yinan He, Jiashuo Yu, Xiangyu Zeng, Chenting Wang, Changlian Ma, Haian Huang, Jianfei Gao, Min Dou, Kai Chen, Wenhai Wang, Yu Qiao, Yali Wang, and Limin Wang. Internvideo2.5: Empowering video mllms with long and rich context modeling. *arXiv preprint arXiv:2501.12386*, 2025. 1

[59] 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. 6

[60] Ye Wang, Boshen Xu, Zihao Yue, Zihan Xiao, Ziheng Wang, Liang Zhang, Dingyi Yang, Wenxuan Wang, and Qin Jin. Timezero: Temporal video grounding with reasoning-guided lvlm. *arXiv preprint arXiv:2503.13377*, 2025. 1, 2, 3, 4, 5, 6

[61] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. In *NeurIPS*, volume 35, pages 24824–24837, 2022. 2

[62] Bo Xiong, Yannis Kalantidis, Deepti Ghadiyaram, and Kristen Grauman. Less is more: Learning highlight detection from video duration. In *CVPR*, pages 1258–1267, 2019. 1

[63] Shen Yan, Xuehan Xiong, Arsha Nagrani, Anurag Arnab, Zhonghao Wang, Weina Ge, David Ross, and Cordelia Schmid. Unloc: A unified framework for video localization tasks. In *ICCV*, pages 13623–13633, October 2023. 6

[64] Dongjie Yang, Suyuan Huang, Chengqiang Lu, Xiaodong Han, Haoxin Zhang, Yan Gao, Yao Hu, and Hai Zhao. Vript: A video is worth thousands of words. In *NeurIPS*, volume 37, pages 57240–57261, 2024. 1

[65] En Yu, Kangheng Lin, Liang Zhao, Jisheng Yin, Yana Wei, Yuang Peng, Haoran Wei, Jianjian Sun, Chunrui Han, Zheng Ge, et al. Perception-r1: Pioneering perception policy with reinforcement learning. *arXiv preprint arXiv:2504.07954*, 2025. 3

[66] Shoubin Yu, Jaemin Cho, Prateek Yadav, and Mohit Bansal. Self-chained image-language model for video localization and question answering. In *NeurIPS*, volume 36, pages 76749–76771, 2023. 1

[67] 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. 3

[68] Xiangyu Zeng, Kunchang Li, Chenting Wang, Xinhao Li, Tianxiang Jiang, Ziang Yan, Songze Li, Yansong Shi, Zhengrong Yue, Yi Wang, Yali Wang, Yu Qiao, and Limin Wang. Timesuite: Improving mllms for long video understanding via grounded tuning, 2024. 6

[69] 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. 3

[70] Xingjian Zhang, Siwei Wen, Wenjun Wu, and Lei Huang. Tinyllava-video-r1: Towards smaller lmms for video reasoning. *arXiv preprint arXiv:2504.09641*, 2025. 3

[71] Yi-Fan Zhang, Xingyu Lu, Xiao Hu, Chaoyou Fu, Bin Wen, Tianke Zhang, Changyi Liu, Kaiyu Jiang, Kaibing Chen, Kaiyu Tang, et al. R1-reward: Training multimodal reward model through stable reinforcement learning. *arXiv preprint arXiv:2505.02835*, 2025. 3

[72] Jiaxing Zhao, Xihan Wei, and Liefeng Bo. R1-omni: Explainable omni-multimodal emotion recognition with reinforcement learning. *arXiv preprint arXiv:2503.05379*, 2025. 1, 3

[73] Hao Zhou, Chongyang Zhang, Yan Luo, Yanjun Chen, and Chuanping Hu. Embracing uncertainty: Decoupling and de-bias for robust temporal grounding. In *CVPR*, pages 8445–8454, 2021. 3

[74] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Yuchen Duan, Hao Tian, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025. 3

Appendix

A Limitations and broader impact

Limitations. While TempSamp-R1 demonstrates consistent improvements over existing GRPO-based methods across multiple temporal grounding benchmarks, it also presents several limitations. First, the framework currently relies on the availability of high-quality off-policy supervision (e.g., ground-truth timestamps), which may not be accessible in weakly labeled scenarios. Second, although TempSamp-R1 is evaluated on temporal grounding and highlight detection tasks, its effectiveness on other video reasoning tasks (e.g., multi-event tracking) remains to be explored.

Broader Impact. This work contributes to the advancement of reinforcement learning in long video understanding. By combining on-policy exploration with structured off-policy guidance, TempSamp-R1introduces a more stable and data-efficient fine-tuning paradigm for vision-language models. We believe this direction can benefit downstream applications such as video retrieval, surveillance, and assistive robotics, where precise temporal reasoning is critical. Since our method is not designed for a specific application domain, it does not directly raise immediate societal or ethical concerns.

B Training details

We fine-tune our method on the Qwen2.5-VL-7B-Instruct model [3] using full-parameter optimization. Tab. 7 summarizes the full set of training configurations. The first-stage fine-tuning uses the same in-domain datasets as the second stage, matching the corresponding evaluation benchmarks (e.g., Charades-STA and QVHighlights).

Trainable	Full Model
Per-device Batch Size	1
Gradient Accumulation	2
Epoch	2
Optimizer	AdamW
Deepspeed	Zero3-Offload
LR Schedule	1×10^{-6}
Max Generated Sequence Length	2048
GPU Nums	4 A100 (80 GB)

Table 7: Training configuration for TempSamp-R1.

C Additional Experiments

Comparison of GRPO and TempSamp-R1 under CoT and non-CoT prompting We compare the performance of GRPO and TempSamp-R1 with both CoT and non-CoT prompting strategies on the Charades-STA dataset. As shown in Tab. 8, TempSamp-R1 consistently outperforms GRPO regardless of the prompting approach. Notably, the improvements achieved with CoT prompting are more pronounced, with gains of 1.5 and 4.8 points at R1@0.5 and R1@0.7 respectively, compared to 1.4 and 3.7 points under non-CoT prompting. These results suggest that the stabilized training and reward optimization in TempSamp-R1 better complement CoT prompting, enabling more effective utilization of intermediate reasoning steps. This highlights the complementary benefits of advanced reward design and CoT prompting in improving temporal grounding.

D Visualizations

We present additional qualitative results to illustrate the behavior of TempSamp-R1 on temporal grounding tasks. Fig. 7 shows representative success cases from both Charades-STA and ActivityNet Captions, where TempSamp-R1 accurately localizes the queried events and generates coherent reasoning under both CoT and non-CoT settings.

In contrast, Fig. 8 highlights common failure cases, which primarily stem from ambiguous visual content, repeated actions, or loosely defined ground-truth annotations. tends to localize the earliest

Method	Training Prompt	Test Prompt	R1@0.3	R1@0.5	R1@0.7
GRPO	Non-CoT	Non-CoT	81.2	70.5	47.1
TempSamp-R1	Non-CoT	Non-CoT	**82.6**	**71.9**	**50.8**
GRPO	CoT	CoT	83.0	72.6	48.1
TempSamp-R1	CoT	CoT	**83.6**	**74.1**	**52.9**

Table 8: Performance comparison of GRPO and TempSamp-R1 with and without CoT prompting on Charades-STA dataset.

matching interval, whereas the annotation provides only a single reference segment, leading to apparent discrepancies. These examples further underscore the challenges of precise temporal localization.

E Datasets of training and evaluation

We train and evaluate TempSamp-R1 on three widely-used video-language datasets, spanning two representative tasks: temporal grounding and highlight detection.

Charades-STA [11] is a benchmark for temporal localization in indoor videos. Each sample consists of a natural language query and its corresponding temporal segment within a video. We follow standard splits with approximately 12.4k training and 3.7k validation examples.

License: Non-commercial use license provided by the Allen Institute for AI. `https://prior.allenai.org/projects/charades`

URL: `https://github.com/jiyanggao/TALL`

ActivityNet Captions [22] provides temporally grounded captions for diverse videos. Each video contains multiple event annotations with rich semantic content. The dataset comprises approximately 20K long, untrimmed videos, with an average of 3.65 sentence-event pairs per video. Following standard practice, we adopt the official dataset splits, resulting in 37,421 training, 17,505 validation, and 17,031 test samples. Compared to Charades-STA, this dataset covers a broader domain and more diverse activity types.

URL: `https://cs.stanford.edu/people/ranjaykrishna/densevid/`

QVHighlights [24] is a large-scale benchmark for evaluating query-conditioned highlight detection in long-form videos. The dataset comprises 10,148 curated video clips, each with a fixed duration of 150 seconds. Each video is paired with at least one user-issued natural language query describing salient moments, resulting in 10,310 unique queries and 18,367 annotated highlights. On average, each highlight spans approximately 24.6 seconds.

License: Attribution-NonCommercial-ShareAlike 4.0 International.

URL: `https://github.com/jayleicn/moment_detr/tree/main/data`

F Prompts

We provide prompt templates used for temporal video grounding and video highlight detection tasks under both CoT and non-CoT settings. For CoT prompts, models are instructed to first reason step-by-step within <think> tags before outputting final predictions in <answer> tags. Non-CoT prompts, by contrast, elicit direct answers without intermediate reasoning. All prompts are designed to standardize response formats and support consistent evaluation across different tasks and inference modes. See Fig. 9 for examples.

Figure 7: Qualitative comparisons on temporal grounding on Charades-STA and ActivityNet Captions. TempSamp-R1 accurately localizes the queried events.

Figure 8: Qualitative analysis of failure cases. Most prediction errors arise from ambiguous visual cues or imperfect ground-truth annotations, such as repetitive events and ill-defined temporal boundaries.

Temporal video grounding (Non-CoT):

"To accurately pinpoint the event "query" in the video, determine the precise time period of the event. Provide the start and end times (in seconds, precise to two decimal places) in the format "start time to end time" within the <answer> </answer> tags. For example: "12.54 to 17.83"."

Temporal video grounding (CoT):

"To accurately pinpoint the event "query" in the video, determine the precise time period of the event. Output your thought process within the <think> </think> tags, including analysis with either specific timestamps (xx.xx) or time ranges (xx.xx to xx.xx) in <timestep> </timestep> tags. Then, provide the start and end times (in seconds, precise to two decimal places) in the format "start time to end time" within the <answer> </answer> tags. For example: "12.54 to 17.83"."

Video highlight detection (Non-CoT):

“Please find the highlight contents in the video described by a sentence query, determining the highlight timestamps and its saliency score on a scale from 1 to 5. Provide the highlight timestamps and its saliency score within the <answer> </answer> tags. The output format should be like: 'The highlight timestamps are in the 82, 84, 86, 88, 90, 92, 94, 96, 98, 100 seconds. Their saliency scores are 1.3, 1.7, 1.7, 1.7, 1.7, 1.3, 1.7, 2.3, 2.3, 2.3'. Now I will give you the sentence query: 'query'. Please return the query-based highlight timestamps and salient scores."

Video highlight detection (CoT):

"Please find the highlight contents in the video described by a sentence query, determining the highlight timestamps and its saliency score on a scale from 1 to 5. Output your thought process within the <think> </think> tags, including analysis with either highlight timestamps. Then, provide the highlight timestamps and its saliency score within the <answer> </answer> tags. The output format should be like: 'The highlight timestamps are in the 82, 84, 86, 88, 90, 92, 94, 96, 98, 100 seconds. Their saliency scores are 1.3, 1.7, 1.7, 1.7, 1.7, 1.3, 1.7, 2.3, 2.3, 2.3'. Now I will give you the sentence query: 'query'. Please return the query-based highlight timestamps and salient scores."

Figure 9: Prompt templates for temporal grounding and video highlight detection tasks.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The claims in the abstract and introduction accurately reflect the contributions, *i.e.* TempSamp-R1 improves the effectiveness of adapting multimodal large language models (MLLMs) to video temporal grounding tasks. The experiments detailed in Sec. 4.1 demonstrate the efficacy of our method in enhancing temporal video grounding, video highlight detection and few-shot performance.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Limitations are discussed in Sec. A.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.

- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: The paper does not include theoretical results.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We give the implementation details and experiment hyperparameters in Sec. 4 and Sec. B. We will make the code, training data and model weights publicly available.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We intend to make our code, training data and model weights publicly available following the paper's acceptance.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: We detail the implementation in Sec. 4 and Sec. B. Additionally, training and testing configurations are included in the provided code.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: Based on our empirical experiments, the reproducibility of results for the same model is very high. The scores we report are reproducible and consistent across multiple trials, rather than being the best scores obtained. Moreover, we face limitations in computational resources, which prevent us from computing error bars for each experiment.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The details regarding computational resources are outlined in Sec. 4 and Sec. B.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research presented in this paper adheres to the NeurIPS Code of Ethics in all aspects.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.

- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: Broader impacts are discussed in Sec. A.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: This paper does not engage with models that necessitate safeguards, and the datasets utilized are publicly available.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

Justification: Licenses of datasets are put into Sec. E.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: This paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: This paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: LLMs were not used as an important, original, or non-standard component of the core methods in this work.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.