# Grounding in the Dark: Text-Only Training for Zero-Shot Video Temporal Localization

**Divyanshu Mishra**[1,2]* **Sabine Sternig**[1] **Rakshith Shetty**[1] **Erhan Gundogdu**[1]
[1]Amazon [2]University of Oxford

## Abstract

Video-LLMs are typically trained on paired video-text data for temporal tasks, yet collecting precise temporal annotations and training on large-scale video datasets are both costly. We investigate whether a text-only LLM, with no vision-language pretraining, can be trained on textual video descriptions alone to perform temporal grounding and receive video only at inference time. This setting removes paired video data from training altogether and is a step toward training multimodal models without paired annotations. We generate a 134K-example corpus of textual sequences with up to 100 time steps, adapt Qwen3-8B with LoRA on CLIP or LLM2CLIP text embeddings, and evaluate on Charades-STA and QV-Highlights without training on videos from either benchmark. LLM2CLIP, whose text encoder supports longer inputs, represents detailed descriptions better than CLIP (61.54 versus 53.86 mtIoU on Temporal-TextVid) and transfers them better to video. Still, replacing text with frame embeddings at inference yields only 4.99 mtIoU on Charades-STA and 11.12 on QV-Highlights, revealing a large modality gap. To bridge it, we propose Nearest Entity Replacement (NER), a training-free mapping that represents each video frame by the normalized mean of its nearest text-entity embeddings, so that the model receives the kind of input it was trained on. NER reaches 14.45 mtIoU on Charades-STA and 20.72 on QV-Highlights, outperforming the strongest of four families of modality-gap baselines by 5.00 and 1.71 points. NER thus enables partial transfer from text to video, while more reliable grounding will likely require encoders with better-aligned text and visual representations.

## 1 Introduction

Video large language models (Video-LLMs) can describe videos, answer questions about them, and localize events in time [9, 21, 34, 40]. Recent trained temporal-grounding variants encode time with dedicated tokens [39], numbered frames [41], sequential transfer from images and short clips [2], or reinforcement-learning objectives [1]. Although their architectures differ, these systems learn from paired video and language. Marking event boundaries requires close inspection by annotators, while storing and repeatedly decoding long videos adds training cost (Fig. 1).

We ask whether a language model can learn a video task from text alone and still operate on video. We start from the Qwen3-8B language checkpoint rather than a pretrained Video-LLM and adapt it with LoRA [8, 43]. A frozen contrastive encoder provides the only pre-existing image-text alignment. During training, its text encoder processes one description per time step; at inference, the corresponding visual encoder processes sampled frames. The downstream projection and LLM remain unchanged. This setup tests whether a temporal model trained on text embeddings can use visual embeddings without retraining.

*Most of this work was done during an internship at Amazon Berlin (August–December 2024).

40th Conference on Neural Information Processing Systems (NeurIPS 2026). Workshop: Grounded and Faithful Vision-Language Models for Real-World Deployment (VLM4RWD).

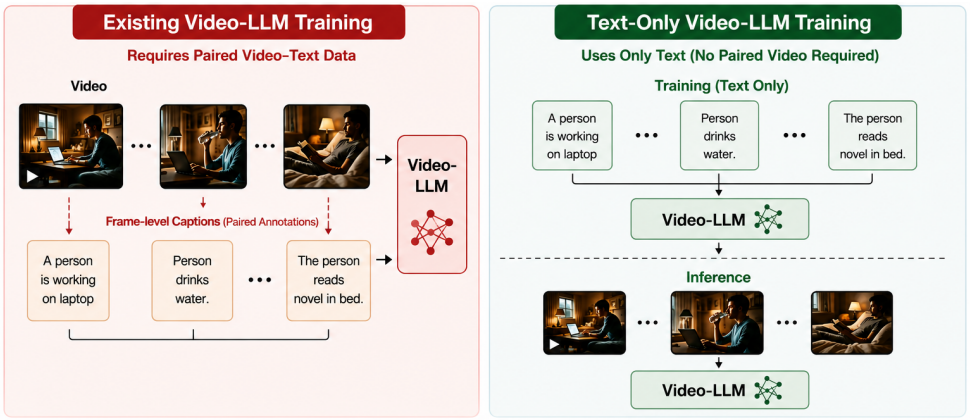



Figure 1: Paired video-text training versus our text-only training and video-inference setting.

Temporal localization provides a demanding transfer test because the query specifies *what* event to find but not *when*. Some VideoQA datasets admit language shortcuts or strong text-only performance [16, 35, 44], whereas our setting replaces every textual position with a frame embedding at inference. The model must find both boundaries among up to 100 positions.

Prior work gives some reason to expect this transfer to work. The Platonic Representation Hypothesis argues that models trained on different modalities recover increasingly similar relational structure as they scale [12], while CLIP explicitly draws matching images and text together [33]. Yet retrieval alignment does not require identical distributions. Multimodal representation spaces have been shown to exhibit a modality gap, with text and visual embeddings retaining modality-specific offsets and geometry despite being mapped into a shared space [26, 48]. As a result, a model trained only on text embeddings may struggle to process visual embeddings, since they occupy different regions of the shared representation space.

The text encoder comes with a separate challenge. CLIP accepts at most 77 text tokens, with an effective context that can be considerably shorter [46]. Even descriptions that fit must compress several objects, actions, and relations into one vector. LLM2CLIP is designed to retain richer language information [11]. Because existing video captions are typically short and at the clip level, we generate detailed and simplified views of the same descriptions to quantify the value of detail and test how existing multimodal encoders represent each.

Our experiments use 134K synthetically generated training examples and evaluate zero-shot performance on Charades-STA [5] and QV-Highlights [19], without using any videos from either benchmark to train the temporal-localization model. We first examine whether the base LLMs can localize events from textual sequences and find that they are capable of doing so (Table 1). The next challenge is transferring this ability to video, where textual descriptions are replaced by visual embeddings. We find that the choice of encoder plays an important role in this transfer. While CLIP struggles with detailed descriptions, LLM2CLIP represents them more effectively, achieving 61.54 mtIoU compared to 53.86 on Temporal-TextVid, and also transfers better to video (Table 2). Based on these results, we adopt LLM2CLIP with detailed descriptions as our main configuration. However, the gap between text and visual representations remains substantial. When evaluated directly on video, the text-trained model achieves only 4.99 mtIoU on Charades-STA and 11.12 mtIoU on QV-Highlights. This motivates us to explore methods for bridging the modality gap. We evaluate four families of approaches and introduce Nearest Entity Replacement (NER), which retrieves fine-grained text entities for each video frame. NER improves performance to 14.45 mtIoU on Charades-STA and 20.72 mtIoU on QV-Highlights (Table 4). While these gains show that the modality gap can be reduced, substantial errors in temporal localization remain. Our contributions are:

- We introduce text-only training for video temporal grounding, using sequences of up to 100 textual time steps. To our knowledge, this is the first work to train a temporal-localization model entirely on text, starting from a text-only LLM rather than a video-aligned one, and transfer it to video without visual fine-tuning or test-time adaptation.

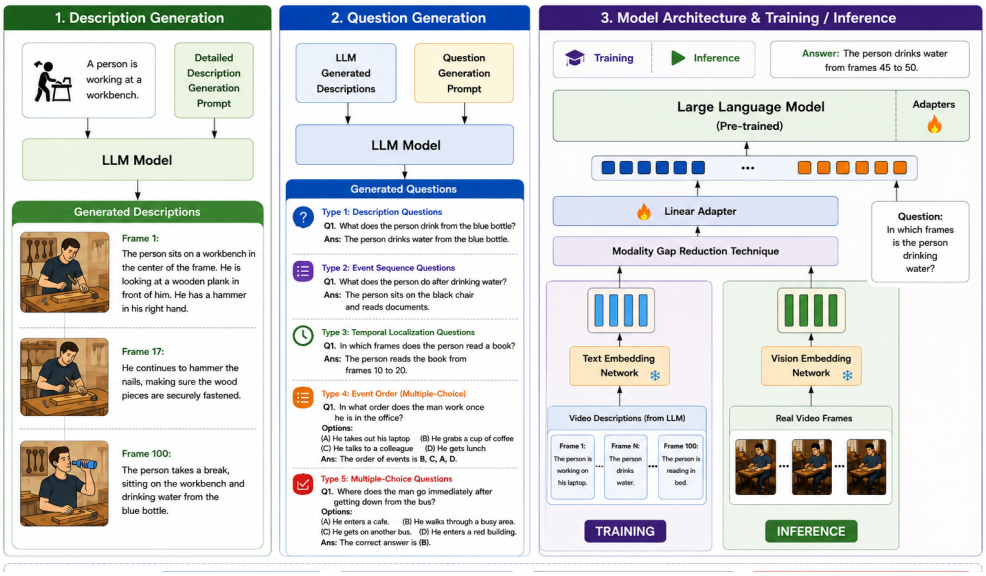



Figure 2: (a) Generation of frame-level textual sequences and questions. (b) Training from text embeddings and inference from video frames, with an optional modality-gap reduction step.

- We study two challenges that arise in text-to-video transfer: preserving fine-grained descriptions in fixed-dimensional embeddings and the modality gap encountered when replacing the text embeddings used during training with visual embeddings at inference.
- We propose Nearest Entity Replacement (NER), a training-free mapping that replaces each frame embedding with the mean of its nearest text-entity embeddings (Eq. 4). Across the matched encoder and caption configurations, NER improves mtIoU over raw visual features and caption-memory projection on both video benchmarks.

## 2 Related Work

**Representation convergence and cross-modal gaps.** Joint embedding models make cross-modal transfer possible by preserving semantic neighborhoods across modalities. CLIP learns this structure through image-text contrastive training [33], and the Platonic Representation Hypothesis places it within a broader trend toward convergent representations across models and modalities [12]. However, semantic agreement does not mean that a downstream model can use raw text and visual embeddings in the same way. Contrastively aligned modalities can occupy distinct regions of the shared space [26]. C3 models part of this difference with a modality-specific mean shift [48], while recent work finds that the remaining gap varies across the embedding space and cannot be corrected with a single average shift [45]. We test this directly by training a temporal model on the text encoder's outputs and applying it to the visual encoder's outputs without retraining.

**Cross-modal learning from a single modality.** Shared spaces have enabled cross-modal tasks to be learned without paired examples for the target task. Noise-injected text training transfers an image captioning decoder from CLIP text embeddings to image embeddings [32], while DeCap first projects image embeddings through a text memory [23]. LinCIR learns composed image retrieval from language alone [6]. The reverse direction is also possible: LAFITE trains text-to-image generation without text by constructing text-space conditions from images [51]. C3 applies modality-specific centering and corruption across captioning and generation tasks [48]. For video, TOPA pre-aligns a base LLM with textual videos before multiple-choice evaluation [24]. T3 [22] instead transfers temporal reasoning learned in text to an already multimodal Video-LLM, and its grounding questions classify an event as occurring at the beginning, middle, or end rather than predicting its boundaries. TD-VAD similarly trains on text sequences and applies the model to CLIP frame features, but for query-free anomaly detection, while concurrent captioning work synthesizes video latents from text

[4, 47]. We instead train a Video-LLM for temporal localization entirely from text sequences and introduce video only at inference.

**Video temporal grounding.** Video-LLMs usually learn temporal grounding from paired video and language, and they differ mainly in how they represent time: discrete temporal tokens with multi-stage video training [39], numbered frames [41], timestamp embeddings with token compression [7], or sequential transfer from images and short clips [2]. Other work improves the training data and objectives through grounding datasets, reinforcement fine-tuning, and comparisons of temporal output formats [1, 14, 25]. A second line of work avoids task-specific annotation. Language-free grounding learns from unlabelled videos and visual pseudo-queries [15], and zero-shot methods apply frozen visual or multimodal models directly to test videos through temporal coherence, visual markers, structured reasoning, self-similarity, or clip-level recognition [3, 10, 13, 17, 18, 20, 29, 42, 49, 50]. Zero-shot temporal action localization targets unseen action classes and may still use video supervision or per-video test-time adaptation [27, 28, 31, 37]. None of these approaches learns the temporal-grounding model itself from text alone.

# 3 Method

## 3.1 Problem formulation and text-only training

For a sequence of $T$ sampled positions, let $D = (d_1, \ldots, d_T)$ denote the frame-level textual descriptions, $q$ a query, and $a = (a_1, \ldots, a_L)$ its answer. In temporal-localization examples, $a$ contains the start and end indices of the queried event. Let $f_T$ and $f_V$ be the text and vision encoders of a pretrained multimodal model, both mapping into $\mathbb{R}^d$. A learned projection $\mathcal{P} : \mathbb{R}^d \to \mathbb{R}^h$ maps their outputs to the token dimension of an LLM $g$.

Our model follows TOPA's text-only pre-alignment architecture [24]. We adapt Qwen3-8B with LoRA [8, 43], using sequences of up to 100 positions for query-conditioned temporal localization. The original backbone weights and multimodal encoders remain frozen, and no Video-LLM checkpoint is used. A trainable linear projection maps each 768-dimensional CLIP or 1,280-dimensional LLM2CLIP feature to the 4,096-dimensional language-model space, and a learned temporal embedding $\boldsymbol{\tau}_t$ marks each position. Training updates the projection, temporal embeddings, and rank-16 LoRA matrices on the query, key, value, and output projections of the attention layers, with LoRA alpha 32 and zero dropout. Features are normalized before the learned projection.

During training, each textual position becomes $\mathbf{h}_t^T = \mathcal{P}(f_T(d_t)) + \boldsymbol{\tau}_t$. The sequence $(\mathbf{h}_1^T, \ldots, \mathbf{h}_T^T)$, followed by the tokenized query, conditions the autoregressive answer:

$$\mathcal{L} = -\sum_{\ell=1}^{L} \log p_g\big(a_\ell \mid \mathbf{h}_{1:T}^T, q, a_{<\ell}\big)\,. \tag{1}$$

At inference, a video $V = (F_1, \ldots, F_T)$ is sampled at the same number of positions and supplies $\mathbf{h}_t^V = \mathcal{P}(f_V(F_t)) + \boldsymbol{\tau}_t$. No parameter is updated at inference. This lets us measure how well a model trained on the outputs of $f_T$ transfers to the outputs of $f_V$.

## 3.2 Synthetic training corpus

To replace video frames with textual descriptions while preserving frame-level details and temporal information, we need a description for each sampled frame rather than a single description for the entire video. Existing video-caption datasets generally describe clips or coarse segments, so we generate Temporal-TextVid from TextVid [24]. As illustrated in Fig. 2(a), Qwen3.8-27B [38] expands each source description into a plausible sequence of up to 100 detailed textual time steps. It also generates questions from five families: description, temporal localization, event ordering, event-order multiple choice, and general multiple choice. Temporal-TextVid contains 134K examples. On average, each example has 29 questions across the five families, giving 3.9M generated questions.

**Caption simplification.** Existing video captions are typically short, so we generate two views of every sequence to quantify the value of detail and to test how existing multimodal encoders represent it. Detailed descriptions preserve more scene information, but they place more content

into a single fixed-dimensional embedding and can exceed the text encoder's input length. For the simplified view, Qwen3.8-27B shortens each detailed description while retaining its principal events and temporal order. Both views therefore contain the same 134K underlying examples rather than two independently sampled corpora. Appendix Fig. 5 shows both views.

### 3.3 Modality-gap reduction baselines

**Gaussian noise injection.** We independently perturb each text embedding during training, $\widetilde{\mathbf{e}}_t^T = f_T(d_t) + \boldsymbol{\epsilon}_t$, where $\boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$. The perturbation exposes the downstream model to randomly modified versions of each text embedding. The same zero-mean Gaussian distribution is used for every caption, so the perturbation does not depend on caption content. No noise is applied at inference. This baseline follows prior single-modality cross-modal training [32, 48, 51].

**Collapse and corrupt.** Let $\mathcal{D}_T$ be the training descriptions and $\mathcal{D}_V$ a reference set of visual inputs. C3 estimates one mean for each modality:

$$\boldsymbol{\mu}_T = \frac{1}{|\mathcal{D}_T|} \sum_{d \in \mathcal{D}_T} f_T(d), \qquad \boldsymbol{\mu}_V = \frac{1}{|\mathcal{D}_V|} \sum_{F \in \mathcal{D}_V} f_V(F). \tag{2}$$

C3 centers and corrupts text during training, $\widetilde{\mathbf{e}}_t^T = f_T(d_t) - \boldsymbol{\mu}_T + \boldsymbol{\epsilon}_t$, and centers visual inputs at inference, $\widetilde{\mathbf{e}}_t^V = f_V(F_t) - \boldsymbol{\mu}_V$ [48]. With a fixed visual reference, the same pair of means is used for every sample, so C3 treats the modality difference as a single dataset-level offset. We compare four estimates of $\boldsymbol{\mu}_V$: the current video, the complete inference split, a similarly sized video dataset, and a large LAION reference sample. Appendix Table 7 reports the full sensitivity analysis.

**Mixed-modality training.** We encode approximately six million LAION-400M images [36] with the vision encoder to form a visual memory. After 35 epochs, the curriculum selects 10% of training examples for replacement. Within a selected example, the variants replace a subset of positions with a nearest visual feature or an average of 10 visual neighbors, or replace every position. These experiments test whether exposure to retrieved visual vectors prepares the LLM for video inference.

**Caption-memory projection.** The DeCap/TOPA baseline [23, 24] stores caption embeddings in a support memory $\mathbf{M}_C = \{\mathbf{c}_i\}_{i=1}^{K_C}$. A visual embedding $\mathbf{v}$ is mapped back to the text space by a temperature-controlled weighted average:

$$\widehat{\mathbf{v}}_C = \sum_{i=1}^{K_C} \frac{\exp(\mathbf{c}_i^\top \mathbf{v}/\lambda)}{\sum_{j=1}^{K_C} \exp(\mathbf{c}_j^\top \mathbf{v}/\lambda)} \mathbf{c}_i. \tag{3}$$

We refer to this as caption-memory projection in the experiments.

### 3.4 Nearest Entity Replacement

NER can be viewed as soft semantic quantization. A frame embedding selects a set of nearby text entities, and their mean replaces the visual vector. The mapping is soft because each frame is represented by an average over many entities rather than one discrete entry. Unlike a single global offset, this replacement depends on the frame content and gives the temporal model the same type of text representation used during training.

**Entity extraction and memory.** A caption-memory entry can bind several concepts into one vector. For example, "a woman is sitting on a chair and working at a table" combines the woman, chair, sitting, working, and table. A frame containing only one of these concepts may retrieve the complete caption and introduce the others (Fig. 3). We call this *concept bleed*: the matching part of a caption brings unrelated caption content into the frame representation. NER instead extracts nouns, verbs, and action phrases from the training descriptions. It encodes each phrase separately with the text encoder and stores the resulting 200K entries in an entity memory $\mathbf{M}_E = \{\mathbf{m}_i\}_{i=1}^{K_E}$.

**Frame-to-entity retrieval.** For each frame $F_t$, NER obtains $\mathbf{v}_t = f_V(F_t)$, selects the $k$ entity vectors with the highest cosine similarity, and records their indices in $\mathcal{N}_k(t)$.

**Aggregation and replacement.** NER replaces the visual vector with the mean of the retrieved text entities:

$$\mathcal{N}_k(t) = \mathrm{TopK}_i \ \frac{\mathbf{v}_t^\top \mathbf{m}_i}{\|\mathbf{v}_t\| \, \|\mathbf{m}_i\|}, \qquad \widehat{\mathbf{v}}_t^{\,E} = \mathrm{Mean}\{\mathbf{m}_i : i \in \mathcal{N}_k(t)\}. \tag{4}$$

The averaging makes the replacement less dependent on any single cross-modal match. The LLM then receives $\{\mathcal{P}(\widehat{\mathbf{v}}_t^{\,E}) + \boldsymbol{\tau}_t\}_{t=1}^{T}$. The visual embedding is used only to retrieve a text-space neighborhood, and no video-side parameter is updated.

# 4 Experimental Setup

**Models and protocol.** All trained temporal models use Qwen3-8B with LoRA and Temporal-TextVid as their training data. CLIP and LLM2CLIP provide the paired text and vision encoders. The main raw-input, caption-projection, and NER configuration uses LLM2CLIP with detailed descriptions. Gaussian, C3, and mixed-modality training use CLIP ViT-L/14 with simplified descriptions, and we also report NER with this CLIP configuration. We evaluate text-side behavior on Temporal-TextVid and video transfer on Charades-STA and the QV-Highlights validation split. No videos from either benchmark are used for training. NER's neighbor count $k$ is selected separately for each encoder and caption view on labelled development videos from each benchmark's training split (Appendix A.5). The main configuration uses $k = 1{,}000$ on Charades-STA and $k = 20$ on QV-Highlights, and NER with CLIP uses $k = 4{,}000$ and $k = 1{,}000$. Training uses 100 epochs, an effective batch size of 1,536, AdamW (peak learning rate $6 \times 10^{-4}$, one warmup epoch, cosine decay), and seed 0. Checkpoints are selected on a separate text monitor.

**Metrics.** Temporal intersection over union (tIoU) divides the intersection of the predicted and reference intervals by their union. We report its dataset mean (mtIoU) and the percentage of examples reaching tIoU thresholds of $0.1$, $0.3$, and $0.5$. All reported values are on a 0–100 scale. We use the archived preprocessing and all-row scoring (Appendix A.5): Charades-STA has 2,572 query–interval examples, and QV-Highlights averages over 2,701 query–reference-window rows (1,486 queries) rather than the best reference per query. Published rows retain their original protocols.

# 5 Results

## 5.1 Temporal localization from text

Before replacing text with video, we test whether frozen Qwen3-8B, prompted without LoRA, and Qwen3.8-27B can localize events directly from Temporal-TextVid descriptions. Each prompt either labels descriptions with frame indices or presents them in order without explicit indices. This isolates temporal reasoning from losses introduced by the embedding encoder and the text-to-video switch. Since Qwen3.8-27B generated the corpus, its score measures consistency with the synthetic descriptions rather than independent generalization.

Frame indices are important for both models (Table 1). Removing them lowers Qwen3-8B's mtIoU from 45.47 to 13.82 and Qwen3.8-27B's from 55.69 to 28.72. This suggests that neither model can reliably identify frame positions from sequence order alone.

## 5.2 Effect of caption detail

We next compress each description into one embedding and compare CLIP with LLM2CLIP (Table 2).

For CLIP, simplifying the descriptions raises Temporal-TextVid mtIoU from 53.86 to 56.64, although both views contain the same examples. The detailed view places more objects, actions, and relations into each fixed-dimensional representation. LLM2CLIP instead scores higher with detailed descriptions (61.54 versus 59.34), consistent with its LLM-based text encoder accepting longer inputs than CLIP's 77 tokens [11].

On video, CLIP fails with detailed descriptions: its raw transfer falls from 7.02 to 2.64 on Charades-STA and from 9.25 to 3.29 on QV-Highlights. LLM2CLIP is much less sensitive to the caption view, moving from 5.61 and 10.83 to 4.99 and 11.12. We therefore keep detailed descriptions with

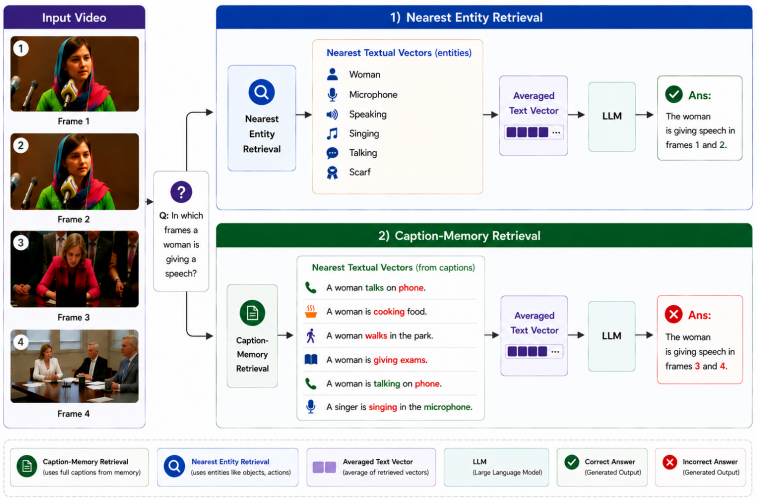



Figure 3: NER avoids concept bleed from unrelated caption content (red).

Table 1: Direct text localization with and without frame IDs on Temporal-TextVid.

| Model | Frame IDs | mtIoU | R@0.1 | R@0.3 | R@0.5 |
|---|---|---|---|---|---|
| Qwen3-8B | Yes | 45.47 | 64.71 | 57.91 | 41.48 |
| Qwen3-8B | No | 13.82 | 25.81 | 20.46 | 11.47 |
| Qwen3.8-27B | Yes | 55.69 | 68.44 | 64.88 | 53.84 |
| Qwen3.8-27B | No | 28.72 | 42.02 | 36.89 | 27.02 |

Table 2: Caption-view and embedding-model ablation (mtIoU). TTV denotes Temporal-TextVid.

| View | Embedding | TTV | Char. | QV-H. |
|---|---|---|---|---|
| Detailed | CLIP | 53.86 | 2.64 | 3.29 |
| Detailed | LLM2CLIP | 61.54 | 4.99 | 11.12 |
| Simplified | CLIP | 56.64 | 7.02 | 9.25 |
| Simplified | LLM2CLIP | 59.34 | 5.61 | 10.83 |

LLM2CLIP, which gives the best text score and avoids CLIP's failure on video. Even so, these scores fall far below its 61.54 text score, highlighting the substantial domain gap.

### 5.3 Improving text-to-video transfer

Table 4 compares NER with four families of methods for closing the text-to-video gap. The training-side baselines use CLIP. Gaussian noise injection and C3 alter the text embeddings independently of frame content and reach only 5.46 and 5.10 mtIoU on Charades-STA, and C3 is also sensitive to the visual reference used for its mean (Table 3). Mixing retrieved image features into training helps more and gives the strongest training baselines, with 9.45 mtIoU on Charades-STA for partial and 12.74 on QV-Highlights for full replacement (Appendix Table 6). At inference, caption-memory projection raises the detailed LLM2CLIP model from 4.99 to 7.95 mtIoU on Charades-STA and from 11.12 to 19.01 on QV-Highlights.

NER outperforms every baseline on both benchmarks. With detailed LLM2CLIP, it reaches 14.45 mtIoU on Charades-STA and 20.72 on QV-Highlights, 9.46 and 9.60 points above raw visual features and 6.50 and 1.71 above caption-memory projection, whose R@0.1 and R@0.3 it more than doubles on Charades-STA. With CLIP, NER reaches 14.33 and 21.85, 4.88 and 9.11 points above the strongest CLIP training baseline.

We attribute the gain over caption projection to the retrieval unit. A caption binds several concepts into one vector, so a frame that matches part of a caption also inherits the rest (Fig. 3). Averaging many single-concept entities instead gives a vector that follows the frame content while staying in the text distribution seen during training. Across encoder and caption configurations, simplified LLM2CLIP + NER scores highest, at 15.57 and 22.29 (Appendix Table 9). However, models trained on paired video-text data still achieve up to 36.80 mtIoU under their original protocols, highlighting the performance gap between single-modality and paired-data approaches.

Table 3: C3 sensitivity to the visual-mean reference with CLIP (mtIoU).

| Reference | Char. | QV-H. |
|---|---|---|
| Current video | **5.78** | **11.55** |
| Evaluation split | 5.09 | 10.76 |
| Other benchmark | 4.74 | 7.31 |
| LAION (5.99M) | 5.10 | 10.16 |

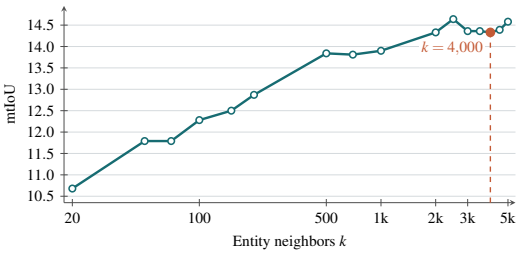



Figure 4: NER (CLIP) on Charades-STA across neighbor counts, with the selected $k = 4{,}000$ marked.

Table 4: Transfer results and published context under original protocols. CLIP rows use simplified descriptions and LLM2CLIP rows detailed descriptions. †: values reproduced from VTimeLLM [9].

| Dataset | Training | Method | Encoder | mtIoU | R@0.1 | R@0.3 | R@0.5 |
|---|---|---|---|---|---|---|---|
| Charades | Video trained | VideoChat-7B† [21] | – | 6.50 | – | 9.00 | 3.30 |
| | | VideoChatGPT-7B† [30] | – | 13.70 | – | 20.00 | 7.70 |
| | | VTimeLLM-7B [9] | – | 31.20 | – | 51.00 | 27.50 |
| | | Grounded-VideoLLM (Phi3.5, 4B) [39] | – | 36.80 | – | 54.20 | 36.40 |
| Charades | Text + image memory | Mixed, full | CLIP | 8.78 | 22.78 | 9.33 | 4.39 |
| | Text only | Gaussian noise injection [32] | CLIP | 5.46 | 17.19 | 3.69 | 1.01 |
| | | C3 [48] | CLIP | 5.10 | 15.36 | 3.46 | 1.05 |
| | | Raw visual features | LLM2CLIP | 4.99 | 12.99 | 5.75 | 2.29 |
| | | Caption projection [23, 24] | LLM2CLIP | 7.95 | 18.93 | 9.18 | 3.97 |
| | | NER (ours) | CLIP | 14.33 | 36.39 | **20.53** | **7.08** |
| | | NER (ours) | LLM2CLIP | **14.45** | **40.01** | 20.49 | 6.07 |
| QV-Highlights | Text + image memory | Mixed, full | CLIP | 12.74 | 29.62 | 15.81 | 9.29 |
| | Text only | Gaussian noise injection [32] | CLIP | 11.14 | 26.95 | 14.18 | 7.74 |
| | | C3 [48] | CLIP | 10.16 | 25.51 | 12.88 | 6.85 |
| | | Raw visual features | LLM2CLIP | 11.12 | 24.51 | 14.07 | 8.03 |
| | | Caption projection [23, 24] | LLM2CLIP | 19.01 | 39.76 | 24.21 | 15.59 |
| | | NER (ours) | CLIP | **21.85** | **45.65** | **26.69** | **17.18** |
| | | NER (ours) | LLM2CLIP | 20.72 | 44.17 | 26.10 | 16.14 |

### 5.4 Sensitivity to the entity-neighbor count

The neighbor count $k$ sets how many entities are averaged for each frame, and larger neighborhoods average out individual wrong matches. For CLIP + NER on Charades-STA (Fig. 4), mtIoU rises from 10.68 at $k = 20$ to 14.33 at the selected $k = 4{,}000$ and varies by less than 0.4 points beyond $k = 2{,}000$, so performance is stable over a wide range of $k$. Development and confirmation results for every configuration are in Appendix A.5.

## 6 Discussion and Limitations

Contrastive training aligns image and text embeddings well enough for retrieval, but our experiments show that this alignment does not make them interchangeable. A model that localizes events well from text loses most of its accuracy when frame embeddings replace the descriptions, even with the stronger LLM2CLIP encoder.

We believe NER works because the temporal model never directly processes visual embeddings. Instead, frame embeddings are used to retrieve nearby text entities, allowing the model to operate on the type of input it was trained on. This lets NER benefit from the retrieval capabilities of existing encoders without requiring their visual and text embedding spaces to be perfectly aligned. In contrast, Gaussian noise and C3 apply the same adjustment to every input regardless of frame content, while C3 also depends on its visual reference. This is consistent with evidence that the modality gap varies across the embedding space [45]. Caption-memory projection is frame-dependent but can introduce caption information that is not present in the frame.

NER is still better at identifying the general region of an event than its precise boundaries, leaving several directions for improvement. Entity averaging could be weighted by similarity or use an adaptive neighborhood for each frame. The mapping could also be learned from unpaired frames and text, while the synthetic corpus could better reflect the temporal pacing of real videos. Better-aligned encoders may eventually enable direct transfer without retrieval. Finally, evaluation on more diverse benchmarks is needed to determine how well these findings generalize.

## References


[1] Ruizhe Chen, Tianze Luo, Zhiting Fan, Heqing Zou, Zhaopeng Feng, Guiyang Xie, Hansheng Zhang, Zhuochen Wang, Zuozhu Liu, and Huaijian Zhang. Datasets and recipes for video temporal grounding via reinforcement learning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 983–992. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-industry.66. URL `https://aclanthology.org/2025.emnlp-industry.66/`.

[2] Andong Deng, Zhongpai Gao, Anwesa Choudhuri, Benjamin Planche, Meng Zheng, Bin Wang, Terrence Chen, Chen Chen, and Ziyan Wu. Seq2Time: Sequential knowledge transfer for video LLM temporal grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13766–13775, 2025. doi: 10.1109/CVPR52734.2025.01285.

[3] Pengcheng Fang, Yuxia Chen, and Xiaohao Cai. MarkIt: Training-free visual markers for precise video temporal grounding. *arXiv preprint arXiv:2604.25886*, 2026. doi: 10.48550/arXiv.2604.25886.

[4] Liangyu Fu, Junbo Wang, Yuke Li, Ya Jing, Xuecheng Wu, and Zhiyong Wang. Watching synthetic videos: Aligning cross-modal representations with visual synthesis for zero-shot video captioning. *arXiv preprint arXiv:2608.11013*, 2026. doi: 10.48550/arXiv.2608.11013.

[5] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. TALL: Temporal activity localization via language query. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5277–5285, 2017. doi: 10.1109/ICCV.2017.563.

[6] Geonmo Gu, Sanghyuk Chun, Wonjae Kim, Yoohoon Kang, and Sangdoo Yun. Language-only efficient training of zero-shot composed image retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13225–13234, 2024. doi: 10.1109/CVPR52733.2024.01256.

[7] 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 *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 3302–3310, 2025. doi: 10.1609/aaai.v39i3.32341.

[8] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

[9] Bin Huang, Xin Wang, Hong Chen, Zihan Song, and Wenwu Zhu. VTimeLLM: Empower LLM to grasp video moments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14271–14280, 2024. doi: 10.1109/CVPR52733.2024.01353.

[10] Ji Huang, Barry Devereux, and Hui Wang. Your VLM already knows when: Training-free temporal grounding by asking yes or no. *arXiv preprint arXiv:2608.08315*, 2026. doi: 10.48550/arXiv.2608.08315.

[11] Weiquan Huang, Aoqi Wu, Yifan Yang, Xufang Luo, Yuqing Yang, Usman Naseem, Chunyu Wang, Qi Dai, Xiyang Dai, Dongdong Chen, Chong Luo, Lili Qiu, and Liang Hu. LLM2CLIP: Powerful language model unlocks richer cross-modality representation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 5131–5139, 2026. doi: 10.1609/aaai.v40i7.37427.

[12] Minyoung Huh, Brian Cheung, Tongzhou Wang, and Phillip Isola. Position: The Platonic representation hypothesis. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 20617–20642. PMLR, 2024. URL `https://proceedings.mlr.press/v235/huh24a.html`.

[13] Mingyu Jeon, Sunjae Yoon, Jonghee Kim, and Junyeong Kim. GranAlign: Granularity-aware alignment framework for zero-shot video moment retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 5287–5295, 2026. doi: 10.1609/aaai.v40i7.37444.

[14] Shengji Jin, Yuanhao Zou, Victor Zhu, Zhengping Ji, and Chen Chen. How should video LLMs output time? An analysis of efficient temporal grounding paradigms. *arXiv preprint arXiv:2604.08966*, 2026. doi: 10.48550/arXiv.2604.08966. CVPR 2026 Workshop paper.

[15] Dahye Kim, Jungin Park, Jiyoung Lee, Seongheon Park, and Kwanghoon Sohn. Language-free training for zero-shot video grounding. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2538–2547, 2023. doi: 10.1109/WACV56688.2023.00257.

[16] Sena Korkut, María Alejandra Bravo Sarmiento, Sanghwan Kim, and Zeynep Akata. From accuracy to visual dependence: Auditing and filtering modality collapse in traffic VideoQA. *arXiv preprint arXiv:2606.30220*, 2026. doi: 10.48550/arXiv.2606.30220.

[17] Jihyun Lee, Cheol-Ho Cho, Woojin Jun, Woojin Jeong, and Jae-Pil Heo. Mitigating modality and language-style gaps for zero-shot video moment retrieval. *arXiv preprint arXiv:2607.19027*, 2026. doi: 10.48550/arXiv.2607.19027. Accepted at ECCV 2026.

[18] Jin-Seop Lee, SungJoon Lee, Jaehan Ahn, YunSeok Choi, and Jee-Hyong Lee. TAG: A simple yet effective temporal-aware approach for zero-shot video temporal grounding. *arXiv preprint arXiv:2508.07925*, 2025. doi: 10.48550/arXiv.2508.07925.

[19] Jie Lei, Tamara L. Berg, and Mohit Bansal. Detecting moments and highlights in videos via natural language queries. In *Advances in Neural Information Processing Systems*, volume 34, pages 11846–11858, 2021.

[20] Boyang Li, Chenhui Gou, and Jianfei Cai. REZE: Recognition-based zero-shot extraction for video temporal grounding. *arXiv preprint arXiv:2608.04480*, 2026. doi: 10.48550/arXiv.2608.04480.

[21] 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.

[22] Lei Li, Yuanxin Liu, Linli Yao, Peiyuan Zhang, Chenxin An, Lean Wang, Xu Sun, Lingpeng Kong, and Qi Liu. Temporal reasoning transfer from text to video. In *International Conference on Learning Representations*, 2025. URL `https://openreview.net/forum?id=sHAvMp5J4R`.

[23] Wei Li, Linchao Zhu, Longyin Wen, and Yi Yang. DeCap: Decoding CLIP latents for zero-shot captioning via text-only training. In *International Conference on Learning Representations*, 2023.

[24] Wei Li, Hehe Fan, Yongkang Wong, Mohan Kankanhalli, and Yi Yang. TOPA: Extending large language models for video understanding via text-only pre-alignment. In *Advances in Neural Information Processing Systems*, volume 37, pages 5697–5738, 2024. doi: 10.52202/079017-0185.

[25] Yunheng Li, Jing Cheng, Shaoyong Jia, Hangyi Kuang, Shaohui Jiao, Qibin Hou, and Ming-Ming Cheng. TempSamp-R1: Effective temporal sampling with reinforcement fine-tuning for video LLMs. *arXiv preprint arXiv:2509.18056*, 2025. doi: 10.48550/arXiv.2509.18056.

[26] Victor Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Y. Zou. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. In *Advances in Neural Information Processing Systems*, volume 35, pages 17612–17625, 2022.

[27] Benedetta Liberatori, Alessandro Conti, Paolo Rota, Yiming Wang, and Elisa Ricci. Test-time zero-shot temporal action localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18720–18729, 2024. doi: 10.1109/CVPR52733.2024.01771.

[28] Benedetta Liberatori, Alessandro Conti, Lorenzo Vaquero, Paolo Rota, Yiming Wang, and Elisa Ricci. Zero-shot temporal action localization through textual guidance. In *2026 IEEE 20th International Conference on Automatic Face and Gesture Recognition*, pages 1–9, 2026. doi: 10.1109/FG67764.2026.11557050.

[29] Dezhao Luo, Jiabo Huang, Shaogang Gong, Hailin Jin, and Yang Liu. Zero-shot video moment retrieval from frozen vision-language models. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5452–5461, 2024. doi: 10.1109/WACV57701.2024.00538.

[30] Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. Video-ChatGPT: Towards detailed video understanding via large vision and language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pages 12585–12602, 2024. doi: 10.18653/v1/2024.acl-long.679.

[31] Sauradip Nag, Xiatian Zhu, Yi-Zhe Song, and Tao Xiang. Zero-shot temporal action detection via vision-language prompting. In *European Conference on Computer Vision*, pages 681–697. Springer, 2022. doi: 10.1007/978-3-031-20062-5_39.

[32] David Nukrai, Ron Mokady, and Amir Globerson. Text-only training for image captioning using noise-injected CLIP. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4055–4063, 2022. doi: 10.18653/v1/2022.findings-emnlp.299.

[33] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pages 8748–8763. PMLR, 2021.

[34] Shuhuai Ren, Linli Yao, Shicheng Li, Xu Sun, and Lu Hou. TimeChat: A time-sensitive multimodal large language model for long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14313–14323, 2024. doi: 10.1109/CVPR52733.2024.01357.

[35] Arka Sadhu, Kan Chen, and Ram Nevatia. Video question answering with phrases via semantic roles. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2460–2478, 2021. doi: 10.18653/v1/2021.naacl-main.196.

[36] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. LAION-400M: Open dataset of CLIP-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021.

[37] Onkar Kishor Susladkar, Gayatri Sudhir Deshmukh, Vandan Gorade, and Sparsh Mittal. GRIZAL: Generative prior-guided zero-shot temporal action localization. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 19046–19059. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.emnlp-main.1061. URL `https://aclanthology.org/2024.emnlp-main.1061/`.

[38] Qwen Team. Qwen3. 8-max: A new bar for coding and cowork. *Qwen release announcement*, 2026.

[39] 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. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 959–975. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-emnlp.50. URL `https://aclanthology.org/2025.findings-emnlp.50/`.

[40] 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.

[41] Yongliang Wu, Xinting Hu, Yuyang Sun, Yizhou Zhou, Wenbo Zhu, Fengyun Rao, Bernt Schiele, and Xu Yang. Number it: Temporal grounding videos like flipping manga. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13754–13765, 2025. doi: 10.1109/CVPR52734.2025.01284.

[42] Yifang Xu, Yunzhuo Sun, Benxiang Zhai, Ming Li, Wenxin Liang, Yang Li, and Sidan Du. Zero-shot video moment retrieval via off-the-shelf multimodal large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 8978–8986, 2025. doi: 10.1609/aaai.v39i9.32971.

[43] An Yang, Anfeng Li, Baosong Yang, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[44] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Just ask: Learning to answer questions from millions of narrated videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1666–1677, 2021. doi: 10.1109/ICCV48922.2021.00171.

[45] Xiaomin Yu, Yijiang Li, Yuhui Zhang, Hanzhen Zhao, Yue Yang, Hao Tang, Yue Song, Xiaobin Hu, Chengwei Qin, Shuicheng Yan, and Hui Xiong. Anisotropic modality align. *arXiv preprint arXiv:2605.07825*, 2026. doi: 10.48550/arXiv.2605.07825.

[46] Beichen Zhang, Pan Zhang, Xiaoyi Dong, Yuhang Zang, and Jiaqi Wang. Long-CLIP: Unlocking the long-text capability of CLIP. In *European Conference on Computer Vision*, pages 310–325. Springer, 2024. doi: 10.1007/978-3-031-72983-6_18.

[47] Shuangqing Zhang, Lei-Lei Ma, Zhao Wang, Wen Dong, Xinyi Xu, Guo-Sen Xie, Caifeng Shan, and Fang Zhao. TD-VAD: Breaking visual dependence in video anomaly detection with text-driven learning. *arXiv preprint arXiv:2608.11820*, 2026. doi: 10.48550/arXiv.2608.11820. Accepted at ICML 2026.

[48] Yuhui Zhang, Elaine Sui, and Serena Yeung-Levy. Connect, collapse, corrupt: Learning cross-modal tasks with uni-modal data. In *International Conference on Learning Representations*, 2024.

[49] Zhengbo Zhang, Mark He Huang, Zhigang Tu, and Ming-Hsuan Yang. DART: Difficulty-adaptive routing for zero-shot video temporal grounding. *arXiv preprint arXiv:2607.00672*, 2026. doi: 10.48550/arXiv.2607.00672. Accepted at ECCV 2026.

[50] Minghang Zheng, Xinhao Cai, Qingchao Chen, Yuxin Peng, and Yang Liu. Training-free video temporal grounding using large-scale pre-trained models. In *European Conference on Computer Vision*, pages 20–37. Springer, 2024. doi: 10.1007/978-3-031-73007-8_2.

[51] Yufan Zhou, Ruiyi Zhang, Changyou Chen, Chunyuan Li, Chris Tensmeyer, Tong Yu, Jiuxiang Gu, Jinhui Xu, and Tong Sun. Towards language-free training for text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17907–17917, 2022.

# A Additional Results

## A.1 Published QV-Highlights Video-LLM results

Video-LLMs commonly evaluate QV-Highlights as highlight detection, using mAP and HIT@1 rather than the interval metrics reported for NER. Table 5 therefore presents these results separately.

Table 5: Published QV-Highlights highlight-detection results. The first three values are reproduced from VTG-LLM [7] and NumPro-FT from Number It [41]. NumPro-FT uses video fine-tuning, and the other rows are zero-shot. These saliency metrics are not directly comparable to NER.

| Method | mAP | HIT@1 |
|---|---|---|
| VideoChat-7B [21] | 13.1 | 18.1 |
| TimeChat-7B [34] | 14.5 | 23.9 |
| VTG-LLM-7B [7] | 16.5 | 33.5 |
| NumPro-FT [41] | 25.0 | 37.2 |

## A.2 Mixed-modality variants

Table 6: Full mixed-modality training results with CLIP.

| Dataset | Variant | mtIoU | R@0.1 | R@0.3 | R@0.5 |
|---|---|---|---|---|---|
| Charades | Partial | 9.45 | 21.77 | 11.86 | 5.56 |
| | Partial + neighbors | 7.07 | 17.19 | 8.20 | 3.65 |
| | Full | 8.78 | 22.78 | 9.33 | 4.39 |
| QV-Highlights | Partial | 12.41 | 26.47 | 15.66 | 9.55 |
| | Partial + neighbors | 9.35 | 22.25 | 11.48 | 6.26 |
| | Full | 12.74 | 29.62 | 15.81 | 9.29 |

## A.3 Sensitivity to the C3 visual mean

C3 requires a visual mean at inference. Table 7 compares a current-video estimate, a transductive estimate from the full evaluation split, a different video dataset, and a 5.99M-image LAION sample. The current-video estimate gives the highest mtIoU on both datasets. LAION is the fixed C3 configuration in Table 4. Neither reference is uniformly best across recall thresholds: on Charades-STA, the evaluation-split estimate has higher R@0.3 and R@0.5 than both. A different video dataset of similar size performs worst on both benchmarks, so the source of the reference matters.

Table 8 reports all recall thresholds for the caption-view and encoder comparison in Table 2.

Table 7: C3 sensitivity to the visual-mean reference with CLIP. The evaluation-split mean uses the full target split and is therefore transductive. The other video set is the other benchmark.

| Dataset | Visual-mean reference | mtIoU | R@0.1 | R@0.3 | R@0.5 |
|---|---|---|---|---|---|
| Charades | Current video | **5.78** | **18.74** | 3.34 | 1.17 |
| | Eval split (133.4K) | 5.09 | 14.74 | **3.50** | **1.28** |
| | Other video set (145.5K) | 4.74 | 14.15 | 2.95 | 0.89 |
| | LAION (5.99M) | 5.10 | 15.36 | 3.46 | 1.05 |
| QV-Highlights | Current video | **11.55** | **28.40** | **15.29** | **7.77** |
| | Eval split (145.5K) | 10.76 | 26.14 | 14.07 | 7.18 |
| | Other video set (133.4K) | 7.31 | 19.62 | 9.07 | 4.07 |
| | LAION (5.99M) | 10.16 | 25.51 | 12.88 | 6.85 |

Table 8: Full caption-view and embedding-model results. TTV denotes Temporal-TextVid.

| View | Embedding | Dataset | mtIoU | R@0.1 | R@0.3 | R@0.5 | R@0.7 |
|---|---|---|---|---|---|---|---|
| Detailed | CLIP | TTV | 53.86 | 69.78 | 65.72 | 51.85 | 43.93 |
| | | Charades | 2.64 | 6.53 | 3.27 | 0.93 | 0.62 |
| | | QV-Highlights | 3.29 | 8.66 | 3.30 | 1.59 | 0.78 |
| Detailed | LLM2CLIP | TTV | 61.54 | 76.18 | 72.68 | 60.18 | 52.31 |
| | | Charades | 4.99 | 12.99 | 5.75 | 2.29 | 1.01 |
| | | QV-Highlights | 11.12 | 24.51 | 14.07 | 8.03 | 4.96 |
| Simplified | CLIP | TTV | 56.64 | 73.18 | 67.95 | 54.39 | 46.27 |
| | | Charades | 7.02 | 16.25 | 8.98 | 4.00 | 2.33 |
| | | QV-Highlights | 9.25 | 20.84 | 11.14 | 6.92 | 4.63 |
| Simplified | LLM2CLIP | TTV | 59.34 | 74.66 | 70.59 | 57.34 | 50.13 |
| | | Charades | 5.61 | 12.79 | 6.42 | 3.30 | 1.98 |
| | | QV-Highlights | 10.83 | 24.44 | 14.55 | 7.92 | 4.44 |

### A.4 Matched encoder and retrieval comparison

Table 9 applies raw features, caption projection (CP), and NER to each text-trained checkpoint. NER, with $k$ selected per configuration on development videos, gives the highest mtIoU in all eight rows.

Table 9: Raw features, caption projection (CP), and NER for each encoder and caption view (mtIoU).

| Dataset | Encoder | View | Raw | CP | NER |
|---|---|---|---|---|---|
| Charades | CLIP | Simplified | 7.02 | 5.31 | **14.33** |
| | CLIP | Detailed | 2.64 | 3.42 | **7.67** |
| | LLM2CLIP | Simplified | 5.61 | 9.37 | **15.57** |
| | LLM2CLIP | Detailed | 4.99 | 7.95 | **14.45** |
| QV-Highlights | CLIP | Simplified | 9.25 | 8.85 | **21.85** |
| | CLIP | Detailed | 3.29 | 6.23 | **15.28** |
| | LLM2CLIP | Simplified | 10.83 | 19.89 | **22.29** |
| | LLM2CLIP | Detailed | 11.12 | 19.01 | **20.72** |

### A.5 Implementation details

**Model and training.** The temporal backbone is Qwen3-8B with frozen weights. LoRA (rank 16, alpha 32, dropout 0) is applied to the query, key, value, and output projections. The linear feature projection (768-dimensional CLIP or 1,280-dimensional LLM2CLIP features to 4,096 dimensions) and the temporal embeddings are trained, and features are normalized before the projection. We train for 100 epochs with batch size 1,536 using AdamW (learning rate $6 \times 10^{-4}$, weight decay 0.1) with one warmup epoch and cosine decay, and select the checkpoint on a held-out text set.

**Retrieval.** The entity memory has 200K entries and the caption memory 500K entries. Caption projection uses $\lambda = 0.01$ and keeps its original settings.

**Neighbor selection.** Model weights stay fixed, and only NER's $k$ is selected with video. For each benchmark, we use 200 source videos from the training split for development and a disjoint 200 for confirmation. Neither overlaps the reported cohorts. For each encoder and caption view, we choose the $k$ with the highest development mtIoU from 16 values between 20 and 5,000. Table 10 compares the selected values with the earlier $k = 3{,}000$ (Charades-STA) and $k = 2{,}000$ (QV-Highlights). Intervals are 95% paired bootstrap intervals over source videos.

Table 10: NER neighbor counts selected on development videos (mtIoU). Reported: full evaluation sets. Confirmation: 200 separate videos per benchmark, with a 95% bootstrap interval for the change.

| Dataset | Encoder | View | $k$ | Reported | Confirmation | Conf. 95% CI |
|---|---|---|---|---|---|---|
| Charades | CLIP | Simplified | 3,000→4,000 | 14.36→14.33 | 12.67→13.42 | [−0.24, 1.73] |
| | CLIP | Detailed | 3,000→5,000 | 7.64→7.67 | 6.63→7.86 | [0.54, 2.02] |
| | LLM2CLIP | Simplified | 3,000→500 | 15.48→15.57 | 16.46→16.39 | [−2.25, 2.06] |
| | LLM2CLIP | Detailed | 3,000→1,000 | 13.91→14.45 | 14.07→15.37 | [−0.31, 2.97] |
| QV-Highlights | CLIP | Simplified | 2,000→1,000 | 21.20→21.85 | 20.93→22.02 | [−0.35, 2.56] |
| | CLIP | Detailed | 2,000→5,000 | 15.52→15.28 | 16.00→14.55 | [−3.12, 0.18] |
| | LLM2CLIP | Simplified | 2,000→100 | 18.63→22.29 | 17.53→21.80 | [1.99, 6.82] |
| | LLM2CLIP | Detailed | 2,000→20 | 16.00→20.72 | 16.71→22.70 | [3.39, 8.78] |

**Evaluation.** The Temporal-TextVid evaluation contains about 2K examples, Charades-STA 2,572 query–interval examples, and QV-Highlights 2,701 query–window rows from 1,486 queries. Scores average over all rows, and the rare unparseable predictions count as zero.

**Compute.** All experiments ran on AWS p5.48xlarge instances with eight NVIDIA H100 80GB GPUs each. Generating Temporal-TextVid with Qwen3.8-27B (FP8, vLLM) took about five hours on six such instances. Each temporal model trained on four GPUs for 20 to 26 hours, about 830 GPU-hours for the nine training runs, and the 216 evaluations, including the neighbor-selection sweeps, took about 39 GPU-hours on single GPUs.

### A.6 Broader impact

Text-only training can reduce paired-video data collection. Predicted intervals can still be incorrect, so downstream systems require task-specific reliability evaluation.

# Generated Captions Example:

## Detailed Caption:

A wide, static shot captures a pristine wilderness setting with towering evergreen trees in the background and distant snow-capped mountains under a clear blue sky. In the foreground, a person wearing a bright blue short-sleeved shirt and khaki shorts is kneeling on the dirt ground, holding a white tent pole with red stripes.

## Simplified Caption:

Person kneels on dirt, holding white tent pole with red stripes, under blue sky near evergreens.

## Entities:

**Visual Entity:** "person", "blue shirt", "khaki shorts", "white tent pole", "red stripes", "evergreen trees", "snow-capped mountains", "blue sky"

**Actions:** "kneeling", "holding"

Figure 5: Detailed and simplified caption views for one Temporal-TextVid example.

## NeurIPS Paper Checklist

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 abstract and Introduction state the text-only setting, evaluation scope, and numerical comparisons reported in the Results section without claiming parity with paired-video training.

   Guidelines:

   - The answer [N/A] 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 [N/A] 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: The Discussion and Limitations section identifies the remaining transfer gap and the two-benchmark scope.

   Guidelines:

   - The answer [N/A] 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: [N/A]

Justification: The paper is empirical and makes no theoretical claims requiring proofs.

Guidelines:

- The answer [N/A] 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: The paper specifies the method, models, LoRA configuration, training hyperparameters, benchmark protocol, memory sizes, neighbor counts, and metrics.

   Guidelines:

   - The answer [N/A] 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: The paper provides provides sufficient evidence to reproduce the paper.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.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://neurips.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) necessary to understand the results?

   Answer: [Yes]

   Justification: The Method and Experimental Setup sections specify the backbone, LoRA configuration, trained parameters, optimizer, schedule, batch size, epochs, seed, checkpoint selection, encoders, neighbor counts, and evaluation protocol, and Appendix A.5 lists further training, retrieval, and evaluation settings, including the development-set selection of NER's neighbor counts.

   Guidelines:

   - The answer [N/A] 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: [Yes]

   Justification: The paper provides the estimates where applicable like Appendix A.5 reports paired source-video bootstrap 95% intervals for the NER neighbor-selection comparisons.

   Guidelines:

   - The answer [N/A] 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 paper does not report hardware type, runtime, or total compute.

   Guidelines:

   - The answer [N/A] 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 study uses established research benchmarks, reports empirical comparisons without concealed deployment claims.

   Guidelines:

   - The answer [N/A] 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: [Yes]

    Justification: Appendix A.6 discusses reduced paired-video data collection and the need for task-specific reliability evaluation of predicted intervals.

    Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] 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., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The submission does not release a model, generated corpus, or other high-risk asset.

    Guidelines:

    - The answer [N/A] 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: Existing models and datasets are credited through their primary citations

    Guidelines:

    - The answer [N/A] 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: [N/A]

    Justification: The fixed synthetic corpus is used as an experimental artifact and is not introduced or released as a new public asset.

    Guidelines:

    - The answer [N/A] 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: [N/A]

    Justification: The study does not involve crowdsourcing or human-subject experiments.

    Guidelines:

    - The answer [N/A] 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: [N/A]

    Justification: The study does not involve human subjects.

    Guidelines:

    - The answer [N/A] 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 rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: The Synthetic Training Corpus section identifies Qwen3.8-27B and describes its role in generating temporal descriptions and questions.

    Guidelines:

    - The answer [N/A] 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 in the NeurIPS handbook for what should or should not be described.