# CONE: An Efficient COarse-to-fiNE Alignment Framework for Long Video Temporal Grounding

**Zhijian Hou**[1]*, **Wanjun Zhong**[2]*, **Lei Ji**[3], **Difei Gao**[4], **Kun Yan**[5], **Wing-Kwong Chan**[1], **Chong-Wah Ngo**[6], **Mike Zheng Shou**[4] and **Nan Duan**[3]

[1] City University of Hong Kong [2] Sun Yat-sen University
[3] Microsoft Research Asia [4] National University of Singapore
[5] Beihang University [6] Singapore Management University

{zjhou3-c@my.,wkchan@}cityu.edu.hk; zhongwj25@mail2.sysu.edu.cn; {leiji,nanduan}@microsoft.com
difei.gao@vipl.ict.ac.cn; kunyan@buaa.edu.cn; cwngo@smu.edu.sg; mike.zheng.shou@gmail.com

## Abstract

This paper tackles an emerging and challenging problem of long video temporal grounding (VTG) that localizes video moments related to a natural language (NL) query. Compared with short videos, long videos are also highly-demanded but less explored, which brings new challenges in higher inference computation cost and weaker multi-modal alignment. To address these challenges, we propose CONE, an efficient COarse-to-fiNE alignment framework. CONE is a plug-and-play framework on top of existing VTG models to handle long videos through a sliding window mechanism. Specifically, CONE (1) introduces a query-guided window selection strategy to speed up inference, and (2) proposes a coarse-to-fine mechanism via a novel incorporation of contrastive learning to enhance multi-modal alignment for long videos. Extensive experiments on two large-scale long VTG benchmarks consistently show both substantial performance gains (e.g., 3.13% $\xrightarrow{+119\%}$ 6.87% on MAD) and state-of-the-art results. Analyses also reveal higher efficiency as the query-guided window selection mechanism accelerates inference time by 2x on Ego4D-NLQ and 15x on MAD while keeping SOTA results. Codes have been released at https://github.com/houzhijian/CONE.

## 1 Introduction

Video temporal grounding (Anne Hendricks et al., 2017; Gao et al., 2017) aims to locate specific moments in an untrimmed video relevant to a textual user query. This is a crucial task in multi-modal video understanding and can be applied to many practical applications such as video retrieval (Xu et al., 2016), video editing, and video question answering (Lei et al., 2018, 2020a).

Temporal grounding for **long videos** is especially highly-demanded and emerging due to the flourishing growth of online videos in quantity and length. However, limited by previously available datasets (Gao et al., 2017; Krishna et al., 2017; Lei et al., 2021), existing works (Zhang et al., 2020a,b) mainly deal with relatively short videos ranging from 0.5 to 2.5 minutes on average. Recently, Ego4D (Grauman et al., 2022) and MAD (Soldan et al., 2022) datasets have been created and attempted to deal with long video, which spans from several minutes to hours.

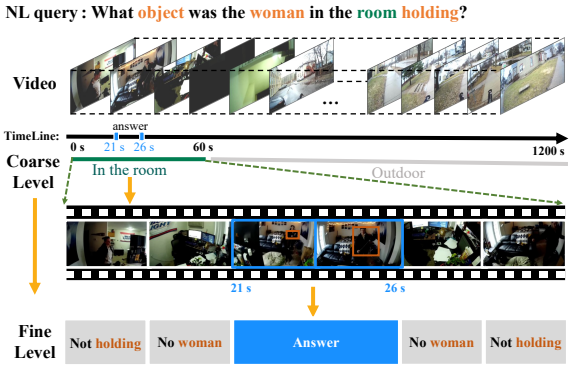


Figure 1: An example of long video temporal grounding, which requires coarse-to-fine multi-modal alignment.

Early attempts for long video grounding extends existing VTG models for short videos through sparse sampling (Grauman et al., 2022) or a sliding window adaptation (Soldan et al., 2022). On the one hand, existing models for short videos typically downsample videos to a fixed-length frame sequence, which leads to *temporal information loss* for long videos (i.e., fewer visible frames via sparse sampling). Besides, the window-based adaptation methods divide the long video into candidate windows and inference on every window, which leads to high inference computational cost. On the other hand, massive moment candidates from long videos make their precise multi-modal alignment with the NL query more challenging, which leads to *contextual information loss* (i.e., weaker alignment to fine-grained contents, like objects, actions, and

* Indicates equal contribution. This work was done during the first and second authors' internships in MSR Asia.



*Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*
*Volume 1: Long Papers*, pages 8013–8028

scenes). As the motivating example shown in Fig. 1, accurate grounding requires a coarse-grained localization of relevant video segments (e.g. "*in the room*" v.s. "*outdoor*"), and fine-grained alignment to detailed object and action in frames (e.g., "*women*" and "*holding*"). In conclusion, long video length poses two new challenges: (1) higher computational cost during inference on the numerous windows of the entire long video; (2) weaker multi-modal alignment due to the abundance of moment candidates.

To address these challenges, we propose CONE, a COarse-to-fiNE alignment framework for long video temporal grounding. First, we slice the arbitrary long video into candidate windows. Then, we employ a query-guided window selection strategy for efficiency and further introduce a coarse-to-fine mechanism for effectiveness. Specifically, the *query-guided window selection* strategy efficiently reduces the sizeable window number of the long video to a modest amount via efficient alignment score computation. Moreover, the *coarse-to-fine* alignment mechanism consists of three modules to gradually align the multi-modal inputs via multi-scale granularity: (1) window (coarse granularity) selection to reliably select semantically relevant candidate windows; (2) window (coarse)-proposal (fine) joint contrastive learning to generate candidate proposals considering both inter-window and intra-window relations; (3) proposal (fine granularity) ranking to further accurately sort out the perfect proposal. Notably, we incorporate contrastive learning into this mechanism. On the one hand, we utilize the powerful multi-modal alignment ability of contrastive vision-text pre-trained models (e.g., CLIP), which computes matching scores between video frames and textual query for both window selection and proposal ranking stages. On the other hand, we select a contrasting negative window and design an inter-window contrastive loss during training for the joint learning stage.

With this coarse-to-fine design, CONE has the following advantages: (1) higher efficiency in handling long video inputs with less temporal information loss; (2) more accurate multi-modal alignment with less contextual information loss. We evaluate CONE on two large-scale long video grounding benchmarks and achieve both the state-of-the-art results and significant performance boosts (3.13%→6.87% on MAD, and 10.46%→13.46% on Ego4D-NLQ in terms of R1@IoU=0.3). Further analysis shows that the window selection strategy shortens the inference speed by 2x for Ego4D-NLQ and 15x for MAD compared to inference on every window, without sacrificing its performance.

**Contributions** Our work presents two major contributions to the long video temporal grounding field. (1) We propose a novel coarse-to-fine alignment framework that utilizes a pipeline of {window slicing and selection, proposal generation and ranking}, resulting in state-of-the-art performance and high efficiency on two representative benchmarks. (2) We introduce a novel approach for incorporating contrastive learning into multi-modal alignment.

## 2 Related Work

**Video Temporal Grounding.** Current models for this task typically fall into two categories based on the usage of proposals. *Proposal-free methods* predict start/end timestamps directly, bypassing the generation of proposals (Ghosh et al., 2019; Zeng et al., 2020; Zhang et al., 2021c). Conversely, *Proposal-based methods* merge proposal generation and ranking within an end-to-end framework (Zhang et al., 2020b; Lei et al., 2021; Cao et al., 2021). An excellent survey paper provides further details on these models (Zhang et al., 2022). However, these models are predominantly designed for relatively short videos, which leads to substantial information loss when adapted directly to long-form settings. Regarding long-form video grounding, VSLNet-L (Zhang et al., 2021b) extends VSLNet (Zhang et al., 2020a) with a multi-scale split-and-concat mechanism to address performance degradation. Still, the multi-scale mechanism adds computation cost during inference, and the sparse sampling approach suffers from significant temporal information loss for hour-long videos. Authors of Ego4D and MAD adapt 2D-TAN (Zhang et al., 2020b) and VLG-Net (Soldan et al., 2021) into simple window-based baselines, yet these models lack the ability for coarse-to-fine alignment, affecting the final results.

Furthermore, other studies have focused on different issues. For example, NaQ (Ramakrishnan et al., 2023) addresses data scarcity with an effective data augmentation strategy, while DeNet (Zhou et al., 2021) tackles the ground-truth bias problem with a de-coupling and de-bias strategy. There are also other related tasks involving language grounding with video corpus input (Lei et al., 2020b; Hou

et al., 2021) or spatial-temporal output (Yang et al., 2022), but they fall outside the scope of this paper.

**Long-form Video Modeling.** Long-form video modeling is an emerging topic recently investigated in (multi-modal) video understanding, including classification, segmentation, localization, and retrieval. The common challenges of long videos are modeling long-range temporal dependency, efficiency issues, and accurate multi-modal alignment (if language is involved). To ease long-range dependency issues, some works explore either feature memory banks (Wu et al., 2019; Feng Cheng, 2022), tracked object-level representations (Wu and Krahenbuhl, 2021), bayesian non-parametric model (Qiu et al., 2023), or structured state-space sequence layers (Islam and Bertasius, 2022). To alleviate efficiency issues, other works (Lin et al., 2022b) exploit dense audio sampling as additional information to enable sparse video frame sampling. In contrast, CONE mitigates both efficiency issues via query-guided window selection strategy and multi-modal alignment issues via novel contrastive learning incorporation.

**Contrastive Learning.** Contrastive learning has been widely studied in vision (Misra and Maaten, 2020; He et al., 2020), language (Gao et al., 2021b; Meng et al., 2021), and multi-modal fields (Liang et al., 2020; Yan et al., 2021; Wang et al., 2022). Some recent VTG works also attempt to adopt contrastive learning. They mainly focus on frame-level contrastive loss via hard negative mining (Zheng et al., 2022) or contrasting ground-truth frame with non-ground-truth frame (Nan et al., 2021; Zhang et al., 2021a). In contrast, we further propose window-level contrastive learning to repel the negative windows. Moreover, CONE also jointly incorporates contrastive vision-text pre-trained models for accurate alignment, because those contrastive models show strong multi-modal alignment ability (Luo et al., 2021; Xu et al., 2021) derived from pre-training large-scale vision-text pairs.

## 3 Task Definition

Since never-ending video streams in real applications have a higher demand for long videos, in this paper, we study the task of video temporal grounding (VTG) in a more challenging setting with long video inputs. Taking a natural language (NL) query $Q$ and a long video $V$ as the inputs, the task of VTG requires the system to locate the matched moment $M$ from the video $V$ relevant to the query $Q$. Formally, video $V = [v_1, v_2, \ldots, v_{L_v}]$ is a sequence of uniformly sampled video frames, and $L_v$ denotes the length of the sampled frames. The NL query $Q = [q_1, q_2, \ldots, q_{L_q}]$ is a sequence of tokens with sentence length $L_q$. The moment $M$ is a subsequence of $V$ that is relevant to $Q$. The long-form video setting is more challenging because of the natural demands for more computation and time to process the entire video. Moreover, the accurate multi-modal alignment between each $v_i$ and $Q$ is also harder when $L_v$ increases.

## 4 Approach

We present the proposed CONE for long video temporal grounding. As shown in Fig. 2, we first slice the long video into several fixed-length video windows via a **sliding window** mechanism (§ 4.1). Then, we propose a coarse-to-fine mechanism for efficient and effective multi-modal alignment. At the coarse-grained window level, we introduce a **query-guided window selection** strategy (§ 4.2) to accelerate inference and select semantic relevant windows. At the window-proposal joint learning level, we rely on the existing VTG work for short videos to generate reliable candidate proposals and conduct both intra-window and proposed **inter-window contrastive learning** (§ 4.3) to assign each proposal score. At the fine-grained proposal level, we further accurately rank candidate proposals with a **fine-grained ranking** (§ 4.4) mechanism.

### 4.1 Window-based Video Slicing

To flexibly handle long videos without decreasing the sample rate and alleviate temporal information loss, we first slice the entire video $V$ into several video windows $W$. A sliding window with window length $L_w$ is used to be slid on the entire video to derive a set of $N_w$ fixed-length video windows $W_i = [v_{w_i^b+1}, v_{w_i^b+2}, ..., v_{w_i^b+L_w}]$, where $w_i^b$ is the start index of window $i$. Concretely, we slide the window by increasing $w^b$ with window stride $L_w/2$. Intuitively, not every window is correlated with the NL query, so we refer the *positive window* to the window overlapping with the ground-truth moment, and the *negative window* otherwise.

### 4.2 Coarse-grained Window Selection

Lengthy video input is sliced into a sequence of windows during inference. If the number of windows is $N_w$, the model needs to conduct the whole encoding-prediction process for $N_w$ times, which

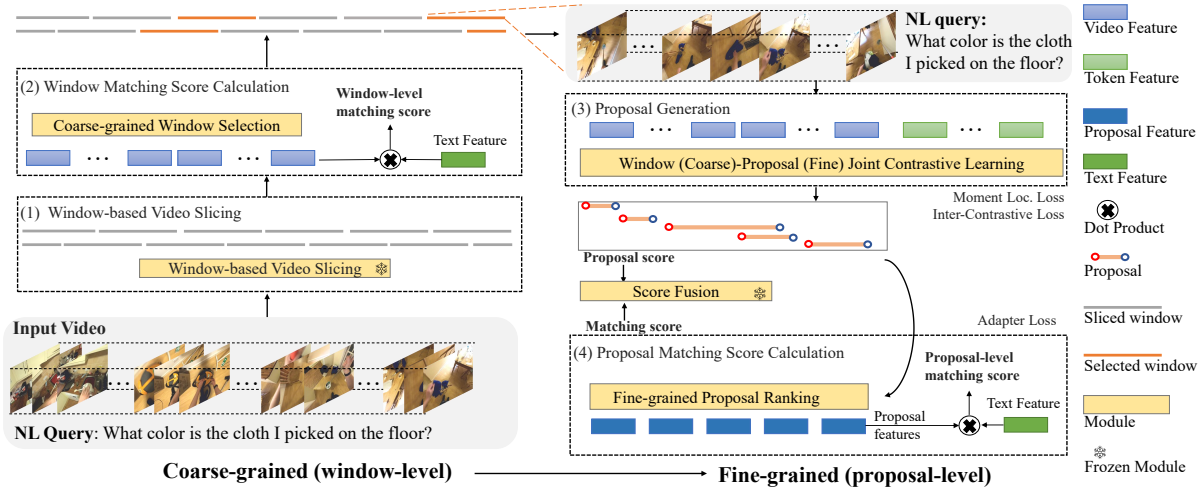


Figure 2: Overview of CONE. It slices the long video into candidate windows (§ 4.1), selects semantic relevant windows (§ 4.2), generates candidate proposals (§ 4.3), and ranks these proposals (§ 4.4).

will become computationally costly with increased video length, especially when the model has enormous parameters. Therefore, it is necessary to reduce the inference computation cost by reliably filtering windows irrelevant to the natural language descriptions. We propose a query-guided window selection strategy via contrastive vision-text pre-trained models.

**Vision-Text Contrastive Model.** CLIP (Radford et al., 2021) and EgoVLP (Lin et al., 2022a) are pre-trained with large-scale vision-text pairs via contrastive learning, aiming to align the visual representation with its related text representation. So they excel at multi-modal alignment, and efficient dot-product score computation provides higher matching efficiency. We utilize the pre-trained model to compute the video features $\boldsymbol{V}$ and the text features $\boldsymbol{Q}$ beforehand.

$$\begin{aligned} \boldsymbol{V} &= [\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_{L_v}] \\ \boldsymbol{Q} &= [\boldsymbol{q}_{\texttt{[CLS]}}, \boldsymbol{q}_1, \boldsymbol{q}_2, \ldots, \boldsymbol{q}_{L_q}] \end{aligned} \tag{1}$$

where [CLS] is a special token at the beginning of text tokens. The multi-modal matching score $a_j = \boldsymbol{v}_j \cdot \boldsymbol{q}_{\texttt{[CLS]}}$ is computed via the efficient dot product between $j^{th}$ video feature and the text feature. And the window-level matching score $A_i$ is the maximum score of all the frames in window $i$:

$$A_i = \max([a_{w_i^b+1}, a_{w_i^b+2}, ..., a_{w_i^b+L_w}]) \tag{2}$$

We rank all windows with $A_i$ and select the top-$k$ windows for inference. Thus, we reduce the number of candidate windows from $N_w$ to a constant $k$ to guarantee a controllable computation cost in an efficient and reliable manner.

### 4.3 Window (Coarse)-Proposal (Fine) Joint Contrastive Learning

Since there are many excellent works in the VTG literature for short video input, we target CONE as a flexible plug-and-play framework on top of the existing proposed-based model. Existing models function as the base model to generate reliable candidate moment proposals for further processing. In our scenario, the base model takes window $W$ and the NL query $Q$ as inputs and outputs several moment proposal candidates. Each of them has a moment proposal $(p_i)$ and corresponding score $(s_i)$, respectively.

Nevertheless, the training of the base model typically focuses only on the positive window and conducts intra-window relation learning but neglects plenty of negative windows for long videos. In real practice, negative windows are the majority during inference, which results in a discrepancy between training and inference. To mitigate the discrepancy, we further design inter-window contrastive learning to distinguish the positive and negative windows. As a result, the overall training loss consists of two parts: (1) the intra-window loss of the base model, and (2) the proposed inter-window contrastive loss.

Concretely, each training instance has both a positive window and a random contrasting negative window from the long video. We discriminate the negative window from the positive window through proposal-level comparison. Proposals in the negative window should be assigned with minimized scores compared with positive proposals (e.g., the IoU with ground truth is large than 0.7) in the posi-

| Dataset | Domain | #Query (train/val/test) | #Video | $N_{vocab}$ | $\overline{L}_{video}$ | $\overline{L}_{query}$ | $\overline{L}_{moment}$ | $\delta_{moment}$ |
|---|---|---|---|---|---|---|---|---|
| Ego4D-NLQ | Open | 11,291 / 3,874 / 4,005 | 998 / 328 / 333 | 3.3K | 8.25min | 7.5 | 8.3s | 2.8s |
| MAD | Movie | 280,183 / 32,064 / 72,044 | 488 / 50 / 112 | 61.4K | 110.77min | 12.7 | 3.9s | 3.2s |

Table 1: Statistics of two benchmarks. $N_{vocab}$ is word vocabulary size, $\overline{L}_{video}$ is average video length, $\overline{L}_{query}$ is average query word number, $\overline{L}_{moment}$ and $\delta_{moment}$ are the average and median ground-truth moment length.

tive window, as follows,

$$\mathcal{L}_{con} = -\sum_{p_i^+ \in W^+} \log(s_i) - \sum_{p_i^- \in W^-} \log(1 - s_i) \tag{3}$$

where $L_{con}$ is the proposed contrastive loss, $p_i^+$ is each positive proposal from the positive window $W^+$ and $p_i^-$ is each proposal from the negative window $W^-$, $s_i$ is the corresponding proposal score.

This mechanism can be generalized to different existing proposal-based models with different intra-window losses. To show the generalization ability of CONE, we test on two kinds of base models: 2D-TAN (Zhang et al., 2020b) and Moment-DETR(Lei et al., 2021) due to their available codes and superior performances. Please refer to Appendix A.2 for comprehensive training details.

### 4.4 Fine-grained Proposal Ranking

With the increased length of video inputs, the fine-grained attention between each video frame and the text query will be weakened by many other perturbed frames, resulting in *contextual information loss*. To remedy this issue, we propose a fine-grained ranking strategy to conduct accurate proposal ranking utilizing proposal matching scores computed by contrastive vision-text pre-trained models (described in § 4.2). Note that we simply re-use the pre-computed video frames and text query features as in Eq. (1).

**Visual Adapter.** With a lightweight visual adapter on top of CLIP, we exploit adapter-based tuning to adapt the representations from the general contrastive model to the data distribution of the current downstream task. Inspired by Gao et al. (2021a), our main idea is to add an additional bottleneck layer to learn the task-adaptive visual features and conduct residual-style blending with the original pre-trained features. The lightweight adapter complies with a 2-layer FFN followed by ReLU. The $i^{th}$ adapted visual feature is: $\hat{\boldsymbol{v}}_i = \text{Adapter}(\boldsymbol{v}_i) + \boldsymbol{v}_i$. Then, the **proposal feature** for the $j^{th}$ proposal is computed with the mean pooling of all the adapted video features in it: $\boldsymbol{h}_j = \text{Mean}([\hat{\boldsymbol{v}}_{b_j}, \ldots, \hat{\boldsymbol{v}}_{e_j}])$

For adapter training, we denote the positive proposal (with feature $\boldsymbol{h}_{pos}$) as the ground-truth one, and the negative proposals are the other in the same batch. We follow the standard contrastive learning and use the NCE loss:

$$\mathcal{L}_{adapt} = -\sum_{pos}(log\frac{\exp(\boldsymbol{h}_{pos} \cdot \boldsymbol{q}_{\texttt{[CLS]}})}{\sum_j \exp(\boldsymbol{h}_j \cdot \boldsymbol{q}_{\texttt{[CLS]}})}) \tag{4}$$

Note that we also use the adapted visual feature to compute the window-level matching score in § 4.2 to improve window selection quality.

**Ranking Score Computation.** Finally, we aim to conduct a fine-grained ranking for proposals. For the $j^{th}$ proposal, the final ranking score is fused with two components: (1) proposal scores $s_j$ generated from the previous module and (2) fine-grained matching scores $m_j$ computed by adapted CLIP-based proposal feature: $m_j = \boldsymbol{h}_j \cdot \boldsymbol{q}_{\texttt{[CLS]}}$. The former captures the correlation between proposals via the sophisticated contextual model design, while the latter focuses on fine-grained content matching between frames in the proposal and the textual query. We perform min-max normalization for these two types of scores for a more stable score fusion. The final ranking score $r_j$ is the sum of two normalized scores:

$$\begin{aligned} \tilde{s_j} &= \textit{MinMax}([s_1, s_2, ..., s_{N_p}]), \\ \tilde{m_j} &= \textit{MinMax}([m_1, m_2, ..., m_{N_p}]) \\ r_j &= \tilde{s_j} + \tilde{m_j} \end{aligned} \tag{5}$$

where $N_p$ is the total candidate proposal number.

## 5 Experiments

We conduct experiments to explore the effectiveness of CONE from the following aspects: (1) model comparison with SOTA methods (§ 5.3); (2) ablation study to analyze the impact of each component and different variants (§ 5.4); (3) efficiency analysis of acceleration with window selection (§ 5.5) and (4) qualitative analysis (§ 5.6). Implementation details are given in **Appendix A.1**.

### 5.1 Dataset

We conduct comprehensive experiments on two representative large-scale benchmarks on long video

| Model | IoU = 0.1 | | | | IoU = 0.3 | | | | IoU = 0.5 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | R1 | R5 | R10 | R50 | R1 | R5 | R10 | R50 | R1 | R5 | R10 | R50 |
| 2D-TAN∗ | 3.22 | 11.90 | 19.01 | 39.63 | 2.52 | 9.25 | 14.72 | 32.65 | 1.58 | 5.69 | 9.06 | 22.46 |
| Moment-DETR∗ | 3.60 | 12.98 | 20.70 | 41.32 | 2.81 | 9.86 | 15.53 | 32.02 | 1.67 | 5.58 | 8.68 | 17.97 |
| VLG-Net | 3.64 | 11.66 | 17.39 | 39.78 | 2.76 | 9.31 | 14.65 | 34.27 | 1.65 | 5.99 | 9.77 | 24.93 |
| CLIP | 6.57 | 15.05 | 20.26 | 37.92 | 3.13 | 9.85 | 14.13 | 28.71 | 1.39 | 5.44 | 8.38 | 18.80 |
| CONE (2D-TAN) | 8.30 | **20.52** | 26.27 | 36.61 | 6.43 | **16.46** | **21.83** | 33.21 | 3.55 | **10.44** | **15.07** | **26.49** |
| CONE (Moment-DETR) | **8.90** | 20.51 | **27.20** | **43.36** | **6.87** | 16.11 | 21.53 | **34.73** | **4.10** | 9.59 | 12.82 | 20.56 |

Table 2: Performance on the test set of MAD dataset. We implement the models with ∗ and report their performances.

| Model | Feature | IoU = 0.3 | | IoU = 0.5 | |
|---|---|---|---|---|---|
| | | R1 | R5 | R1 | R5 |
| Validation split | | | | | |
| 2D-TAN | SF + B | 5.04 | 12.89 | 2.02 | 5.88 |
| Moment-DETR∗ | SF + B | 5.09 | 17.17 | 2.99 | 9.16 |
| 2D-TAN∗ | EgoVLP | 7.10 | 18.28 | 3.82 | 11.00 |
| Moment-DETR∗ | EgoVLP | 8.23 | 23.23 | 5.01 | 13.37 |
| VSLNet | EgoVLP | 10.84 | 18.84 | 6.81 | 13.45 |
| CONE (2D-TAN) | EgoVLP | 11.00 | 25.06 | 6.09 | 15.51 |
| CONE (Moment-DETR) | SF + B | 10.64 | 24.47 | 5.76 | 13.32 |
| CONE (Moment-DETR) | EgoVLP | **14.15** | **30.33** | **8.18** | **18.02** |
| Test split | | | | | |
| 2D-TAN∗ | EgoVLP | 6.89 | 14.86 | 3.80 | 8.47 |
| Moment-DETR∗ | EgoVLP | 9.14 | 18.66 | 5.05 | 10.59 |
| VSLNet | EgoVLP | 10.46 | 16.76 | 6.24 | 11.29 |
| CONE (2D-TAN) | EgoVLP | 11.08 | 19.96 | 5.87 | 11.44 |
| CONE (Moment-DETR) | EgoVLP | **13.46** | **23.68** | **7.84** | **14.16** |

Table 3: Performance on the Ego4D-NLQ dataset. SF + B is short for the SlowFast and Bert features. We implement the models with ∗ and report their performances.

temporal grounding: Ego4D-NLQ (Grauman et al., 2022) and MAD (Soldan et al., 2022). The data statistics are summarized in Table 1.

**Ego4D-NLQ** is a subtask of the Ego4D dataset. Ego4D is a large-scale egocentric video understanding benchmark, where 931 camera wearers worldwide record their daily activities in hundreds of scenarios. The unedited videos involved have variant lengths ranging from 3.5 min. to 20 min. The NL query is designed to retrieve the relevant moment from the episodic memory of camera wearers, and involves 13 question types for locating different types of information.

**MAD** is a large-scale long video temporal grounding benchmark on full-length movie videos. The video duration is magnificent longer than previous benchmarks at an average length of 110.8 min (ranging from 47 min. to 202 min). The textual queries in the training set are derived from translated movie audio descriptions from professional narrators. The annotations in the evaluation set are of higher quality derived from the LSMDC dataset (Rohrbach et al., 2017) with more strict manual refinement. The NL queries are cleaner and the ground-truth moments have preciser temporal boundaries.

## 5.2 Experimental Settings

### 5.2.1 Evaluation Metric.

For consistent comparison, we follow the previous baseline evaluation setting and use the metric Recall@$k$ at IoU $= \theta$ (R@$k$). It is the percentage of queries, having at least one prediction among the top-$k$ predictions, whose temporal IoU with ground-truth is larger than the threshold $\theta$ (0.3 or 0.5). Note that there is only one ground-truth answer for each query in both datasets.

### 5.2.2 Visual and Textual Features.

For MAD dataset, we use CLIP (Radford et al., 2021) to extract the visual and textual features, which are then utilized in the inputs of the window selection, proposal generation, and fine-grained ranking stages. The MAD authors extract the visual features every 0.2s (5fps). For Ego4D-NLQ dataset, we adopt the egocentric contrastive pre-trained model, i.e., EgoVLP (Lin et al., 2022a), to extract visual and textual features, because CLIP is trained with third-person image-text pairs and has the domain adaptation gap for first-person videos. We extract the visual feature every 16 frames (1.875fps).

## 5.3 Model Comparison

### 5.3.1 Baselines.

We compare CONE to the following methods: (1) the window-based adaptation of proposal-based models, i.e., **2D-TAN** (Zhang et al., 2020b), **VLG-Net** (Soldan et al., 2021) and **Moment-DETR** (Lei et al., 2021); (2) the sparse sampling-based proposal-free model **VSLNet** (Zhang et al., 2020a); (3) the two-stage model **CLIP** (Radford et al., 2021), which first generates offline proposals

and then ranks those proposals with CLIP matching scores.

**Results on Ego4D-NLQ.** Table 3 reports the performance comparison on the validation and the blind test set of Ego4D-NLQ. Regarding all the metrics, CONE outperforms these baselines by a large margin. Take R1@IoU=0.3 and R5@IoU=0.3 as examples, the absolute performance gains are +3.31% and +11.49% on the val. set, and +3% and +6.92% on the blind test set, respectively. CONE also achieves consistent performance gains using different features and base models, showing a better generalization ability.

**Results on MAD.** The main results of MAD are shown in Table 2, and demonstrate that CONE achieves state-of-the-art performances with obvious boost (e.g., 3.13%$\xrightarrow{+119\%}$6.87% and 9.85%$\xrightarrow{+63.5\%}$16.11% improvement on R1@IoU=0.3 and R5@IoU=0.3).

From the two tables, the SOTA performance on two benchmarks demonstrates the effectiveness of CONE. We speculate the outstanding results are brought by the following reasons: (1) CONE processes the entire video without decreasing sample rate, which alleviates temporal information loss; (2) the coarse-to-fine mechanism enables the better alignment ability of relevant proposals to the NL query and reduces contextual information loss. The lower results on the MAD benchmark also verify that the grounding task indeed becomes more challenging with longer videos.

Regarding base model selection, we observe that, compared with Moment-DETR, the performance of 2D-TAN is worse in Ego4D-NLQ but comparable in the MAD dataset. This gives the clues that the base model might not be influential to the final results given more training samples for hour-level videos. Because of the lower parameter and GLOPs number of Moment-DETR, from now on, CONE adopts it as the default base model.

### 5.4 Model Analysis

Ablation studies are shown in table 4 to unveil the effectiveness of each component in CONE. We consider three components: (a) **contrastive loss** (§ 4.3), which is eliminated by only training on positive windows without our contrastive loss; (b) **fine-grained ranking fusion** (§ 4.4) can be removed by taking only the proposal score from the proposal generation module for ranking; (c) **visual adapter** (§ 4.4) is removed by using general CLIP-based features for matching score computation. Note that the full CONE model refers to the first row, and the baseline model (i.e., the window-based adaptation of Moment-DETR) refers to the last row.

| Model | Ego4D-NLQ | MAD |
|---|---|---|
| CONE | 14.15 | 6.91 |
| w/o visual adapter | 12.62 | 6.73 |
| w/o fine-grained ranking | 11.59 | 4.66 |
| w/o contrastive learning | 8.23 | 2.56 |

Table 4: Cumulative ablation study on the val. set of Ego4D-NLQ and MAD datasets, taking **R1@IoU=0.3** as the metric.

From the table 4, we highlight the following findings for Ego4D-NLQ dataset (MAD shows similar trends): (1) Ablating visual adapter leads to a performance drop from 14.15% to 12.62%, which indicates that domain-adaptation of visual features is essential in modelling task-specific semantic variance. (2) Eliminating fine-grained ranking also harms the performance (row 2 vs. row 3), showing that capturing fine-grained semantic alignment benefits accurate grounding. (3) Further removing contrastive loss (row 3 vs. row 4) leads to a significant performance drop of 3.36%, which reveals that identifying the inter-window semantic variance is critical for reliable grounding.

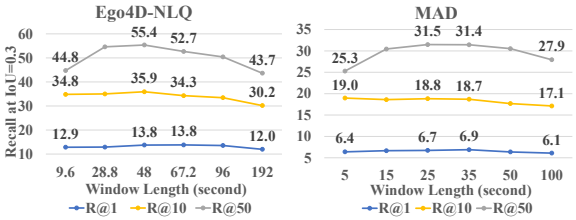


Figure 3: Performance (Recall k@IoU=0.3) on both datasets w.r.t different window lengths.

**Influence of window length.** Figure 3 exhibits how different window lengths affect the overall performance of CONE. We observe that increasing the window length indeed brings performance variance. Longer window length (more visible frames in a single window) can model the interaction between more frames, but can also weaken the multi-modal attention between the NL query and every single frame as the performance drops significantly with the largest window size. We find a better trade-off between window length and performance, i.e., nearly 48 seconds ( 90 video features) for Ego4D-NLQ and nearly 25 seconds (125 video features) for MAD through this analysis.

### 5.5 Efficiency Analysis

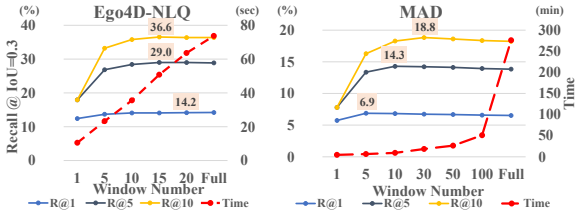


Figure 4: Inference time w.r.t window numbers is denoted by **red dashed** lines for efficiency analysis. Performance (Recall k@IoU=0.3) w.r.t window numbers is shown with solid lines.

Figure 4 shows inference speed with respect to different window numbers after the window selection stage on the val. sets of Ego4D-NLQ and MAD. We observe that the overall inference time is approximately linearly reduced with smaller window numbers, and the performance of CONE is relatively stable when the filtered window number becomes 10 for both datasets. Surprisingly, we also observe that CONE achieves the optimal performance (shown as the box with value in Fig. 4) at a modest size of window number rather than the full size. Those observations show that our query-guided window selection strategy can reliably filter the irrelevant windows and largely improve the efficiency of long video temporal grounding.

Concretely, the average window numbers for full videos (before filtering) are 23.3 and 588 for Ego4D-NLQ and MAD, respectively. If we set the filtered window number to 10 for Ego4D-NLQ and 30 for MAD (better trade-off values obtained from Fig. 4), the total numbers of windows for inference are reduced by 2.3x and 19.6x, with a marginal performance variance (R@1) by -0.1% and +0.2% for Ego4D-NLQ and MAD benchmarks, respectively. In real practice, we consider the effect of implementation methods and window selection time. When CONE is inferred using one P100 GPU, its running time (w/o feature extraction and post-processing time, because both time costs are the same for all methods) is largely reduced (74 s $\xrightarrow{\text{-2x}}$ 36 s and 276 min. $\xrightarrow{\text{-15x}}$ 18 min.) on the val. set of Ego4D-NLQ and MAD.

Furthermore, we also conduct an inference comparison between VSLNet and CONE on Ego4D-NLQ. Using the official Ego4D released code, VSLNet takes approximately 12 seconds to infer the Ego4D validation split and achieves a Recall@1 score of 10.84 at IoU=0.3. In contrast, CONE, with the extreme setting of selecting only the top 1 window, takes about 10.5 seconds to infer the Ego4D validation split and achieves a Recall@1 score of 12.4 at IoU=0.3. It demonstrates that CONE strikes an optimal tradeoff between accuracy and speed compared to sparse sampling-based baselines.

### 5.6 Qualitative Analysis

Fig. 5 shows two success examples to analyze the effect of **contrastive learning** and **fine-grained ranking**. We observe that baseline Moment-DETR (a) is not capable of repelling negative windows and tends to give an equally high score to the proposal in the negative window, thus it wrongly ranks the correct prediction to a much lower position (e.g., $54^{th}$ in Example A). Further adding inter-window contrastive learning mechanism (b), the rank position of the ground truth moment is somewhat improved, but it still lacks fine-grained matching to detailed contents to push the perfect proposal into the first place. For example, in Query-A, the most essential object is the vegetable rather than the bag; In Query-B, the query requires the precise scene “in the park" and the objects “two women" and “ribbons". Finally, adding fine-grained ranking (c), the full CONE successfully locates the ground-truth moment relevant to the textual query. More qualitative examples with both success and failure cases are shown in Appendix B.

## 6 Conclusions

We present CONE , a COarse-to-fiNE alignment framework for long video temporal grounding. CONE jointly achieves state-of-the-art performances on two benchmarks and high efficiency while keeping high frame sample rate. Through our experiments, we show the proposed coarse-to-fine mechanism via contrastive learning plays an important role in boosting performance. Regarding efficiency, the introduced query-guided window selection strategy largely accelerates inference by 2x and 15x on Ego4D-NLQ and MAD benchmarks, respectively, with even slight performance gains. Since CONE can be generalized with different existing proposal-based models, we hope it can be used to improve the model efficiency and performance for temporal grounding on long videos.

## Limitations

Currently, CONE is mainly implemented for proposal-based models as they can generate ex-

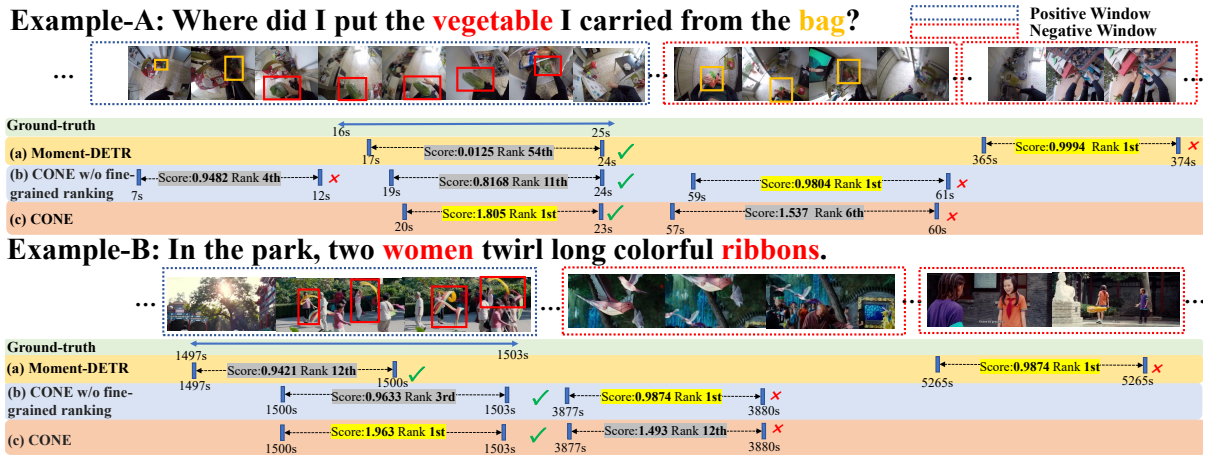


Figure 5: Two successful examples. Example-A and Example-B come from the Ego4D-NLQ and MAD datasets, respectively. We compare three settings: (a) baseline Moment-DETR; (b) CONE w/o fine-grained ranking and (c) the full CONE. The major difference between (a) and (b) is **inter-window contrastive learning**.

plicit moment proposals for the introduced inter-window contrastive learning. In contrast, proposal-free methods (e.g., VSLNet (Zhang et al., 2020a)) directly predict the start/end timestamps without explicit proposals. In the future, it is worthwhile to explore how to incorporate the coarse-to-fine alignment mechanism with proposal-free methods to enhance the generalization ability of CONE.

Furthermore, CONE falls short on the ground-truth moment case whose duration is longer than the adopted video window duration. To ease this issue, future work can explore adaptive-duration window slicing to ensure complete containment of scenes and events within windows or some rule-based proposal merging techniques.

## Ethics Statement

The present study was conducted in accordance with ethical principles. This study involved the analysis using publicly available data and did not involve any human participants, and potential risks about credentials or privacy. Therefore, no ethical clearance was required and there were no potential risks associated with the conduct of this research.

## Acknowledgements

We thank the anonymous reviewers for their insightful feedback. This research was partially supported by CityU MF_EXT project number 9678180.

## References

Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. 2017. Localizing moments in video with natural language. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5803–5812.

Meng Cao, Long Chen, Mike Zheng Shou, Can Zhang, and Yuexian Zou. 2021. On pursuit of designing multi-modal transformer for video grounding. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9810–9823, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Gedas Bertasius Feng Cheng. 2022. Tallformer: Temporal action localization with a long-memory transformer. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. 2017. TALL: temporal activity localization via language query. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5267–5275.

Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. 2021a. Clip-adapter: Better vision-language models with feature adapters. *arXiv preprint arXiv:2110.04544*.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021b. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Soham Ghosh, Anuva Agarwal, Zarana Parekh, and Alexander G. Hauptmann. 2019. Excl: Extractive clip localization using natural language descriptions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 1984–1990.

Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. 2022. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18995–19012.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738.

Zhijian Hou, Chong-Wah Ngo, and Wing Kwong Chan. 2021. Conquer: Contextual query-aware ranking for video corpus moment retrieval. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 3900–3908.

Md Mohaiminul Islam and Gedas Bertasius. 2022. Long movie clip classification with state-space video models. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. 2017. Dense-captioning events in videos. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 706–715.

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

Jie Lei, Licheng Yu, Mohit Bansal, and Tamara Berg. 2018. TVQA: Localized, compositional video question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1369–1379, Brussels, Belgium. Association for Computational Linguistics.

Jie Lei, Licheng Yu, Tamara Berg, and Mohit Bansal. 2020a. TVQA+: Spatio-temporal grounding for video question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8211–8225, Online. Association for Computational Linguistics.

Jie Lei, Licheng Yu, Tamara L Berg, and Mohit Bansal. 2020b. Tvr: A large-scale dataset for video-subtitle moment retrieval. In *Proceedings of the European Conference on Computer Vision*, pages 447–463.

Zujie Liang, Weitao Jiang, Haifeng Hu, and Jiaying Zhu. 2020. Learning to contrast the counterfactual samples for robust visual question answering. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pages 3285–3292.

Kevin Qinghong Lin, Alex Jinpeng Wang, Mattia Soldan, Michael Wray, Rui Yan, Eric Zhongcong Xu, Difei Gao, Rongcheng Tu, Wenzhe Zhao, Weijie Kong, et al. 2022a. Egocentric video-language pretraining. *arXiv preprint arXiv:2206.01670*.

Yan-Bo Lin, Jie Lei, Mohit Bansal, and Gedas Bertasius. 2022b. Eclipse: Efficient long-range video retrieval using sight and sound. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. 2021. Clip4clip: An empirical study of clip for end to end video clip retrieval. *arXiv preprint arXiv:2104.08860*.

Yu Meng, Chenyan Xiong, Payal Bajaj, Paul Bennett, Jiawei Han, Xia Song, et al. 2021. Coco-lm: Correcting and contrasting text sequences for language model pretraining. *Advances in Neural Information Processing Systems*, 34:23102–23114.

Ishan Misra and Laurens van der Maaten. 2020. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6707–6717.

Guoshun Nan, Rui Qiao, Yao Xiao, Jun Liu, Sicong Leng, Hao Zhang, and Wei Lu. 2021. Interventional video grounding with dual contrastive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2765–2775.

Jielin Qiu, Franck Dernoncourt, Trung Bui, Zhaowen Wang, Ding Zhao, and Hailin Jin. 2023. Liveseg: Unsupervised multimodal temporal segmentation of long livestream videos. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5188–5198.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR.

Santhosh Kumar Ramakrishnan, Ziad Al-Halah, and Kristen Grauman. 2023. Naq: Leveraging narrations as queries to supervise episodic memory. *arXiv preprint arXiv:2301.00746*.

Anna Rohrbach, Atousa Torabi, Marcus Rohrbach, Niket Tandon, Christopher Pal, Hugo Larochelle, Aaron Courville, and Bernt Schiele. 2017. Movie description. *International Journal of Computer Vision*, 123(1):94–120.

Mattia Soldan, Alejandro Pardo, Juan León Alcázar, Fabian Caba, Chen Zhao, Silvio Giancola, and Bernard Ghanem. 2022. Mad: A scalable dataset for language grounding in videos from movie audio descriptions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5026–5035.

Mattia Soldan, Mengmeng Xu, Sisi Qu, Jesper Tegner, and Bernard Ghanem. 2021. Vlg-net: Video-language graph matching network for video grounding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3224–3234.

Zixu Wang, Yujie Zhong, Yishu Miao, Lin Ma, and Lucia Specia. 2022. Contrastive video-language learning with fine-grained frame sampling. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 694–705.

Chao-Yuan Wu, Christoph Feichtenhofer, Haoqi Fan, Kaiming He, Philipp Krahenbuhl, and Ross Girshick. 2019. Long-term feature banks for detailed video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 284–293.

Chao-Yuan Wu and Philipp Krahenbuhl. 2021. Towards long-form video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1884–1894.

Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. 2021. VideoCLIP: Contrastive pre-training for zero-shot video-text understanding. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6787–6800, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5288–5296.

Kun Yan, Lei Ji, Huaishao Luo, Ming Zhou, Nan Duan, and Shuai Ma. 2021. Control image captioning spatially and temporally. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2014–2025.

Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. 2022. Tubedetr: Spatio-temporal video grounding with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16442–16453.

Runhao Zeng, Haoming Xu, Wenbing Huang, Peihao Chen, Mingkui Tan, and Chuang Gan. 2020. Dense regression network for video grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Hao Zhang, Aixin Sun, Wei Jing, Guoshun Nan, Liangli Zhen, Joey Tianyi Zhou, and Rick Siow Mong Goh. 2021a. Video corpus moment retrieval with contrastive learning. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 685–695.

Hao Zhang, Aixin Sun, Wei Jing, Liangli Zhen, Joey Tianyi Zhou, and Rick Siow Mong Goh. 2021b. Natural language video localization: A revisit in span-based question answering framework. *IEEE transactions on pattern analysis and machine intelligence*.

Hao Zhang, Aixin Sun, Wei Jing, and Joey Tianyi Zhou. 2020a. Span-based localizing network for natural language video localization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6543–6554, Online. Association for Computational Linguistics.

Hao Zhang, Aixin Sun, Wei Jing, and Joey Tianyi Zhou. 2022. The elements of temporal sentence grounding in videos: A survey and future directions. *arXiv preprint arXiv:2201.08071*.

Mingxing Zhang, Yang Yang, Xinghan Chen, Yanli Ji, Xing Xu, Jingjing Li, and Heng Tao Shen. 2021c. Multi-stage aggregated transformer network for temporal language localization in videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12669–12678.

Songyang Zhang, Houwen Peng, Jianlong Fu, and Jiebo Luo. 2020b. Learning 2d temporal adjacent networks for moment localization with natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12870–12877.

Minghang Zheng, Yanjie Huang, Qingchao Chen, Yuxin Peng, and Yang Liu. 2022. Weakly supervised temporal sentence grounding with gaussian-based contrastive proposal learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15555–15564.

Hao Zhou, Chongyang Zhang, Yan Luo, Yanjun Chen, and Chuanping Hu. 2021. Embracing uncertainty: Decoupling and de-bias for robust temporal grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8445–8454.

# A Model Details

## A.1 Implementation Details

During training, we perform parameter optimization via AdamX and set the learning rate to 1e-4 for the base model and 1e-5 for the visual adapter. We set the batch size to 32 and adopt the early stopping strategy. During inference, we use Non-Maximum Suppression (NMS) with a threshold of 0.5 as post-processing.

**Moment-DETR Base Model.** The experiments are conducted on one P100 GPU. For the hyperparameters of network architecture, we set the hidden size $d$ to 256, the transformer layer number in the encoder/decoder to 2, and the moment query number to 5. As a result, the total parameters of CONE are 4.35M (4.22M Moment-DETR + 0.13M visual adapter). We set the window length to 90 video features (48 seconds) and 125 video features (25 seconds) for the Ego4D-NLQ and MAD datasets, respectively. We train Ego4D-NLQ for 150 epochs and MAD for 30 epochs, and the training time is about 3 hours for the Ego4D-NLQ and 18 hours for the MAD dataset. During inference, we set the filtered window number to 10 and 30 for the Ego4D-NLQ and MAD datasets, respectively.

**2D-TAN Base Model.** The experiments are conducted on two V100 GPUs. For the hyperparameters of network architecture, we set the hidden size $d$ to 256, the convolution network layer to 4, and the kernel size to 9. As a result, the total parameters of CONE are 23.99M (23.86M 2D-TAN + 0.13M visual adapter). We set the window length to 64 video features (34.1 seconds) and 128 video features (25.6 seconds) for the Ego4D-NLQ and MAD datasets, respectively. We train Ego4D-NLQ for 90 epochs and MAD for 6 epochs, and because of 6x more model parameters than Moment-DETR, the training time is quite larger, i.e., about 18 hours for the Ego4D-NLQ and 3 days for the MAD dataset. During inference, we set the filtered window number to 5 and 15 for the Ego4D-NLQ and MAD datasets, respectively.

## A.2 Training Loss Details

**Moment-DETR Base Model.** The original loss of Moment-DETR consists of three parts: moment localization, classification, and saliency losses. Localization loss requires ground truth moment to measure the discrepancy with the predictions and can not be used in contrastive loss. Thus, we design two-level contrastive losses based on two other losses: (1) proposal-level loss and (2) frame-level loss with a randomly sampled negative window.

For the proposal-level contrastive loss ($\mathcal{L}_p$), proposals in the negative window are assigned with minimized scores compared to positive proposal (i.e., the optimal proposal selected from the Hungarian algorithm) from the positive window, as follows,

$$\mathcal{L}_p = -\sum_{p_i^+ \in W^+} \log(s_i) - \sum_{p_i^- \in W^-} \log(1 - s_i) \quad (6)$$

where $p_i^+$ is the positive proposal from the positive window $W^+$ and $p_i^-$ is each proposal from the negative window $W^-$, $s_i$ is the corresponding proposal score.

For the frame-level loss $\mathcal{L}_f$, we set the average saliency scores for frames located in the positive window is larger than the maximum saliency score of frames in the negative window over a margin $\delta$:

$$\mathcal{L}_f = \max(0, \delta + \max(S(W^-, Q)) - \text{mean}(S(W^+, Q)) \quad (7)$$

where S() is its saliency scoring function. So the overall contrastive loss is $\mathcal{L}_{con} = L_p + L_f$.

**2D-TAN Base Model.** The original loss of 2D-TAN consists of the binary cross-entropy loss, which learns to align each proposal score with its scaled IoU value. Thus, we assign each proposal in the negative window with minimized scores compared to the positive proposals (i.e., the IoU with ground truth is large than 0.7) from the positive window, as follows,

$$\mathcal{L}_p = -\sum_{p_i^+ \in W^+} \log(s_i) - \sum_{p_i^- \in W^-} \log(1 - s_i) \quad (8)$$

where $p_i^+$ is the positive proposal from the positive window $W^+$ and $p_i^-$ is each proposal from the negative window $W^-$, $s_i$ is the corresponding proposal score. And the overall contrastive loss is $\mathcal{L}_{con} = L_p$.

**Overall Loss.** In total, our training loss ($\mathcal{L}$) consists of three parts: (1) original training loss of the base model; (2) contrastive loss to discriminate the negative window versus the positive window; (3) adapter loss (shown in Eq. 4) to tune visual representations from general pre-training to the current downstream task:

$$\mathcal{L} = L_{ori} + \lambda_{con} \times L_{con} + \lambda_{adapt} \times L_{adapt} \quad (9)$$

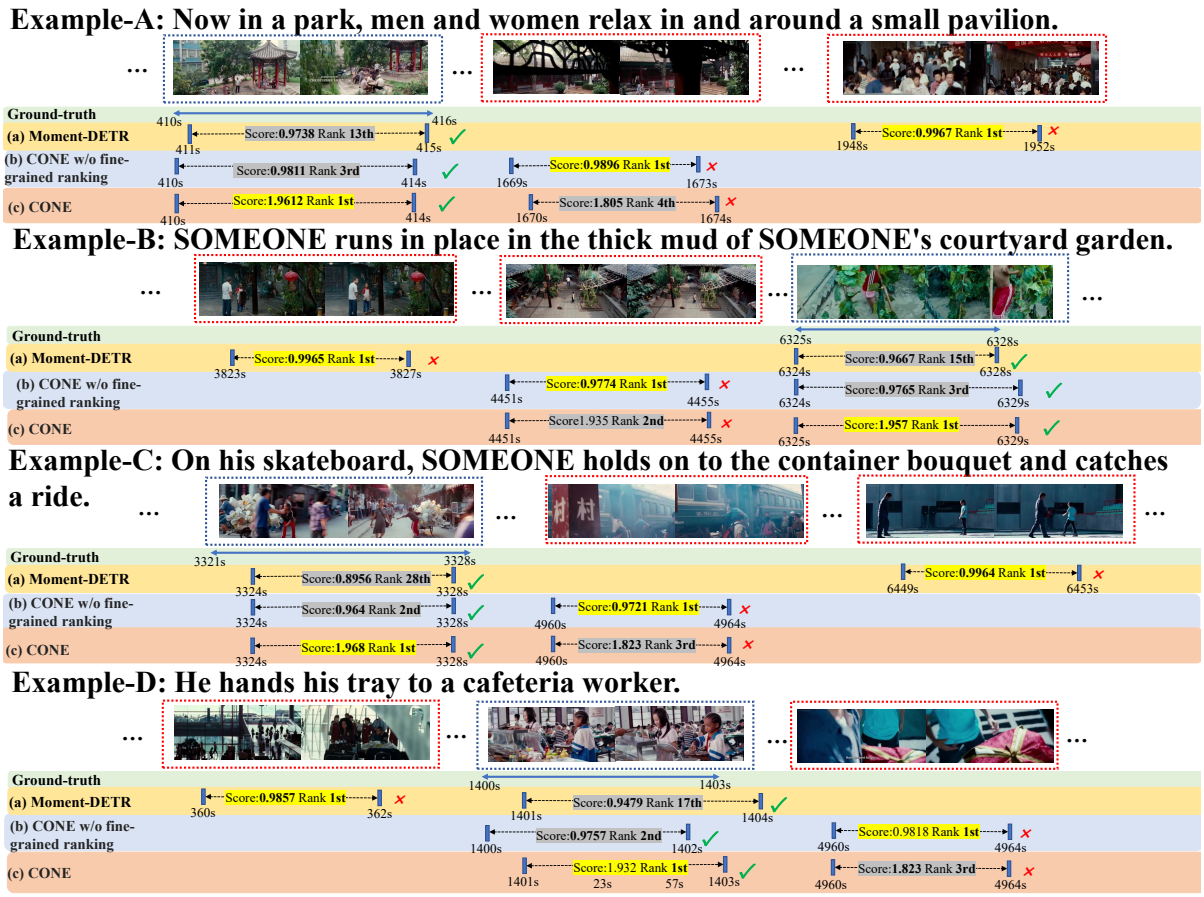


Figure 6: Four success examples on the MAD dataset.

where $\lambda_{con}$ and $\lambda_{adapt}$ are loss weight hyperparameters to control the loss value. During training, we set $\lambda_{con}$ to 1 and $\lambda_{adapt}$ to 0.2.

## B More Qualitative Examples

We show four success cases in Figure 6 and four failure cases of our model CONE in Figure 7.

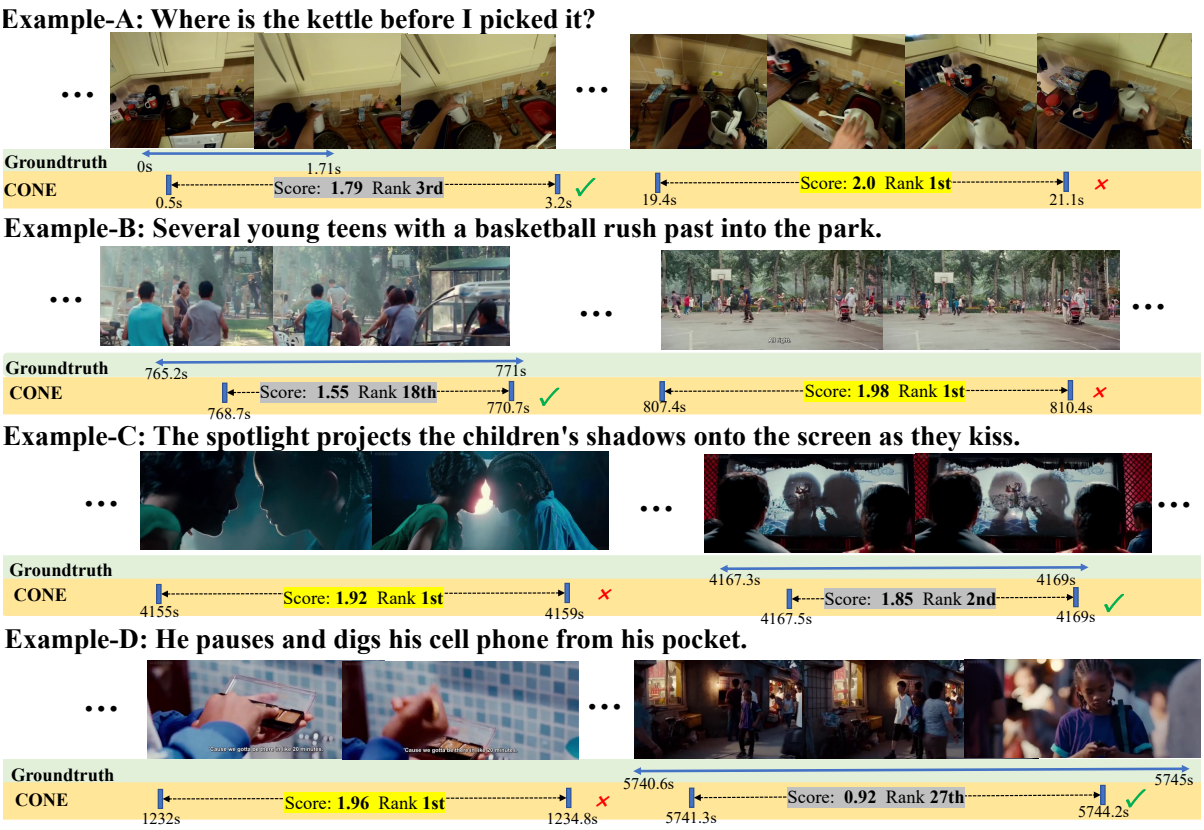


Figure 7: Four failure examples. Example-A comes from the Ego4D-NLQ dataset; the remaining three examples are from the MAD dataset.

## ACL 2023 Responsible NLP Checklist

### A For every submission:

☑ A1. Did you describe the limitations of your work?
*section 6 page 9*

☑ A2. Did you discuss any potential risks of your work?
*section 6 page 9*

☑ A3. Do the abstract and introduction summarize the paper's main claims?
*section 1*

☒ A4. Have you used AI writing assistants when working on this paper?
*Left blank.*

### B ☑ Did you use or create scientific artifacts?

*experiments section 5*

☑ B1. Did you cite the creators of artifacts you used?
*experiments section 5*

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
*Not applicable. Left blank.*

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
*Ethics Statement*

☑ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
*Ethics Statement*

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
*Not applicable. Left blank.*

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
*table 1*

### C ☑ Did you run computational experiments?

*experiments section 5*

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
*appendix A*

*The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.*

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
*appendix A*

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
*experiments section 5*

☐ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
*Not applicable. Left blank.*

**D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?**

*Left blank.*

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
*Not applicable. Left blank.*

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
*Not applicable. Left blank.*

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
*Not applicable. Left blank.*

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
*Not applicable. Left blank.*

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
*Not applicable. Left blank.*