Entity-aware and Motion-aware Transformers for Language-driven Action Localization

Shuo Yang , **Xinxiao Wu***

Beijing Laboratory of Intelligent Information Technology,
School of Computer Science, Beijing Institute of Technology
{shuoyang,wuxinxiao}@bit.edu.cn

Abstract

Language-driven action localization in videos is a challenging task that involves not only visual-linguistic matching but also action boundary prediction. Recent progress has been achieved through aligning language query to video segments, but estimating precise boundaries is still under-explored. In this paper, we propose entity-aware and motion-aware Transformers that progressively localizes actions in videos by first coarsely locating clips with entity queries and then finely predicting exact boundaries in a shrunken temporal region with motion queries. The entity-aware Transformer incorporates the textual entities into visual representation learning via cross-modal and cross-frame attentions to facilitate attending action-related video clips. The motion-aware Transformer captures fine-grained motion changes at multiple temporal scales via integrating long short-term memory into the self-attention module to further improve the precision of action boundary prediction. Extensive experiments on the Charades-STA and TACoS datasets demonstrate that our method achieves better performance than existing methods.

1 Introduction

Language-driven action localization, also called temporal video grounding or video moment retrieval, aims to localize the start and end frames of an action relevant to the language query. It has attracted growing attention for its wide applications, such as robotic navigation and video understanding. This task is challenging since it requires not only aligning the language query to video segments but also estimating the temporal boundaries of the desired action.

Tremendous effects have been devoted to the alignment between language query and video segments. Several early studies [Hendricks *et al.*, 2017] resort to learning a common visual-textual embedding space by pushing dissimilar or pulling similar visual features and linguistic features. Later, in order to explore more detailed semantics for visual-textual alignment, some methods [Chen and Jiang, 2019] extract semantic concepts of actions or objects to enrich the holistic features of both video and language. In more recent years, various attention operations have been proposed to learn elaborate cross-modal relations, such as self-modal or cross-modal graph attention [Liu *et al.*, 2020], context-query attention [Zhang *et al.*, 2021a] and local-global interaction [Mun *et al.*, 2020]. All these methods mainly focus on learning and aligning the visual and linguistic representations for language-driven action localization without considering the explicit modeling of finer action boundaries for precise localization.

Figure 1: Illustration of coarse-to-fine action localization using entity and motion queries.

This paper investigates a coarse-to-fine strategy to progressively estimate the action boundaries in untrimmed videos with high precision.[1] With this in mind, we propose entity-aware and motion-aware Transformers that first coarsely locate video clips from the entire video with textual entities and then finely predict exact boundaries in a shrunken temporal region with motion queries. For example, as illustrated in Figure 1, the query sentence of "*a person is watching the mirror*" can be divided into two types of information: the entities of "*person & mirror*" and the motion of "*is watching*". Our method first finds the frames in which the "*person & mirror*" appear, and then localizes the start and end boundaries between which the "*is watching*" happens.

To be more specific, the entity-aware Transformer incorporates textual entities of language query into visual representation learning via cross-modal attention. The learned visual features are capable of attending to the salient action-related objects so as to facilitate selecting action-related video clips. Moreover, cross-frame attention is employed to leverage con-

*corresponding author

[1]Code is available at https://github.com/shuoyang129/eamat

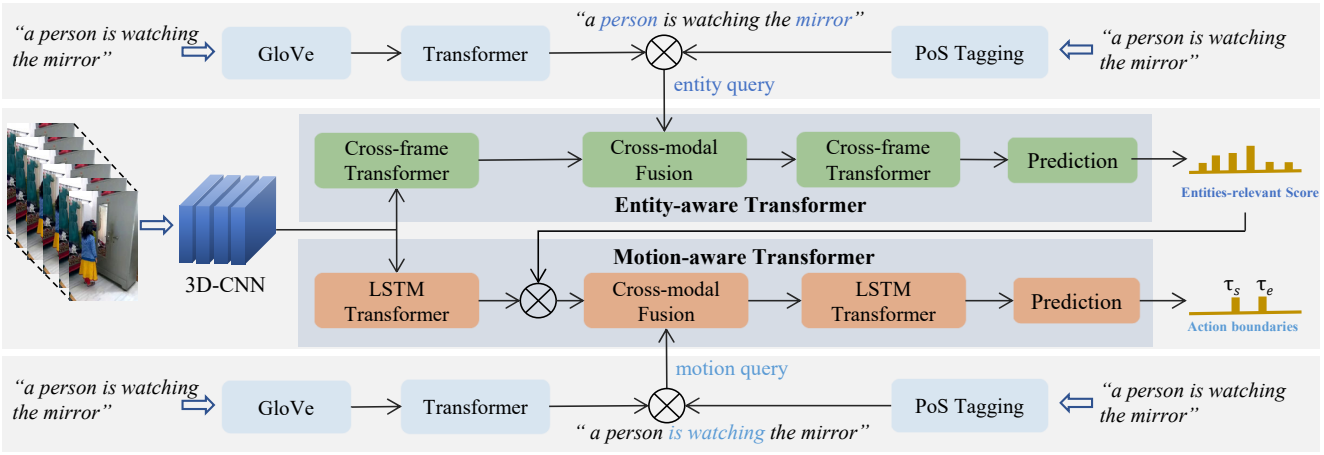


Figure 2: Overview of our entity-aware and motion-aware Transformers for language-driven action Localization.

textual information from adjacent frames to learn more robust entity features. Starting with more accessible entities, the entity-aware Transformer narrows the searching space for action localization by coarsely locating temporal regions where the desired action is more likely to happen.

An action consists of sequential motions and large motion changes usually lie on the action boundaries. To improve the action localization precision, it is significantly essential to capture the fine-grained motion changes in videos. So we propose a motion-aware Transformer that integrates a long short-term memory cell into the self-attention module in Transformer. Intuitively, the long short-term memory cell is a natural way to capture the consecutive local motion changes, and we apply it at multiple temporal scales to deal with various durations of the same action. Transformer is capable of modeling the long-range dependency and has been proved its effectiveness in many visual and linguistic tasks [Vaswani *et al.*, 2017], and it is reasonable to use it for modeling the global motion interactions. Therefore, our motion-aware Transformer can capture fine-grained motion changes at multiple time granularities, which benefits a lot to localizing the exact boundaries of desired actions.

The main contributions of this paper are summarized as follows: (1) We propose a coarse-to-fine framework for language-driven action localization, which extracts detailed entity and motion queries to progressively estimate the action boundaries with high precision. (2) We propose entity-aware and motion-aware Transformers as an effective implement of the coarse-to-fine localization, where the newly designed motion-aware Transformer models fine-grained motion changes at multiple temporal scales by integrating long short-term memory into self-attention. (3) Extensive experiments on popular benchmarks, Charades-STA and TACoS, demonstrate that the proposed method performs favorably against existing methods.

2 Related Work

The language-driven action localization task is firstly proposed in [Gao *et al.*, 2017; Hendricks *et al.*, 2017]. It is tackled by first generating proposals with manually designed temporal bounding boxes and then ranking the proposals by the given language query. To enhance the visual and linguistic representations, ACRN [Liu *et al.*, 2018] proposes a memory attention mechanism to emphasize the language-related visual features with context information. SCDM [Yuan *et al.*, 2020] modulates the temporal convolution operations for better correlating and composing the sentence related video contents. 2D-TAN [Zhang *et al.*, 2020] uses a 2D temporal adjacent network to learn contextual and structural information between adjacent moment candidates. MAST [Zhang *et al.*, 2021c] aggregates multi-stage features to represent moment proposals using a BERT-variant Transformer backbone. These proposal-based methods are relatively inefficient since a large number of proposals causes redundant computation. Moreover, the boundaries of proposals are fixed, leading to inflexible estimations.

To mitigate the defects of manually designed proposals, proposal-free methods [Zeng *et al.*, 2020; Yuan *et al.*, 2019; Hahn *et al.*, 2019; Lu *et al.*, 2019; Wu *et al.*, 2020; Li *et al.*, 2021; Zhao *et al.*, 2021] are proposed to directly predict the action boundaries through visual and linguistic representation alignment. ExCL [Ghosh *et al.*, 2019] and SeqPAN [Zhang *et al.*, 2021b] predict the start and end time by leveraging the cross-modal interaction between the text and video; LGI [Mun *et al.*, 2020], CSMGAN [Liu *et al.*, 2020], FIAN [Qu *et al.*, 2020], CBLN [Liu *et al.*, 2021], SMIN [Wang *et al.*, 2021], I^2N [Ning *et al.*, 2021] explore the local and global context information for accurate localization.

Rather than mainly focusing on aligning the visual and linguistic representations in the aforementioned methods, we attempt to achieve high localization precision by designing a progressive strategy that first narrows the target regions and then localizes the finer boundaries. The most related work to our method is VSLNet [Zhang *et al.*, 2021a] that searches for the target action within a highlighted region, which extends the target action segment by a simple hyper-parameter in a span-based question answering framework. In contrast, our method coarsely locates the temporal region first by the apparent action-related entities, and then finely predicts the

action boundaries by explicitly modeling fine-grained motion changes at both short and long times, achieving high precision, good generalization and interpretability.

3 Our Method

Given an untrimmed video $V = \{v_t\}_{t=1}^{T}$ and a language query $S = \{w_i\}_{i=1}^{N}$ where v_t represents the t-th video frame, w_i represents the i-th word, and T and N represent the number of video frames and text words, respectively, our task aims to localize the target action boundaries (τ_s, τ_e) where τ_s and τ_e represent the start and end frames of the action corresponding to the query, respectively. As shown in Figure 2, our method has two main components: an entity-aware Transformer and a motion-aware Transformer. The former incorporates the entity terms, *i.e.,* subjects and objects, of the language query into the visual representation learning to filter out the video clips that have no action-relevant entities. The latter captures fine-grained motion changes by integrating a long short-term memory cell into the self-attention module guided by the motion terms, *i.e.,* verbs, of the language query to refine the start and end frames.

3.1 Entity and Motion Query Extraction

We encode the input language query S into entity query features $\mathbf{F}_q^e$ and motion query features $\mathbf{F}_q^m$. The words in S are classified into three classes: entity, motion, and others, by using the part of speech tags[2] of the words. The classification probabilities of the word i are denoted by $\mathbf{p}_i = [p_i^e; p_i^m; p_i^o] \in \{0,1\}^3$. For the word i, if its part of speech tag is related to entity (*i.e.*, noun, adjective), then $\mathbf{p}_i = [1,0,0]$; if its part of speech tag is related to motion (*i.e.*,verb, adverb), then $\mathbf{p}_i = [0,1,0]$; otherwise, $\mathbf{p}_i = [0,0,1]$.

The word features $\mathbf{Q} = [\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_N]^\top \in \mathbb{R}^{N \times d_w}$ of S are first initialized using the GloVe embedding [Pennington *et al.*, 2014], where $\mathbf{w}_i$ denotes the i-th word feature with dimension d_w and N denotes the word number in the language query. And then a Transformer block is used to learn the relationships between the words, given by

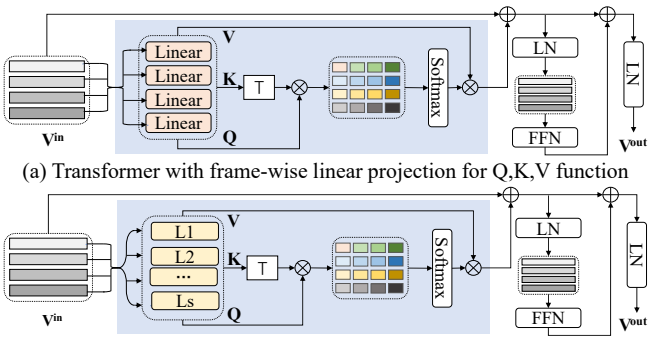
$$\mathbf{F}_q = Transformer_q(FC_1(\mathbf{Q})) \tag{1}$$

where $\mathbf{F}_q = [\mathbf{f}_{q,1}, \mathbf{f}_{q,2}, \cdots, \mathbf{f}_{q,N}]^\top \in \mathbb{R}^{N \times d}$ are the learned linguistic query features; $FC_1(\cdot)$ is a fully connected layer that projects the word feature from dimension d_w to d; $Transformer_q(\cdot)$ is a standard Transformer block, as shown in Figure 3(a), which consists of multi-head self-attention, residual connection, layer normalization and feed-forward network. Finally, the entity query features $\mathbf{F}_q^e = [\mathbf{f}_{q,1}^e, \mathbf{f}_{q,2}^e, \cdots, \mathbf{f}_{q,N}^e]^\top \in \mathbb{R}^{N \times d}$ and motion query features $\mathbf{F}_q^m = [\mathbf{f}_{q,1}^m, \mathbf{f}_{q,2}^m, \cdots, \mathbf{f}_{q,N}^m]^\top \in \mathbb{R}^{N \times d}$ of the language query are calculated by

$$\mathbf{f}_{q,i}^e = \mathbf{f}_{q,i} \cdot (p_i^e + p_i^o), \quad \mathbf{f}_{q,i}^m = \mathbf{f}_{q,i} \cdot (p_i^m + p_i^o). \tag{2}$$

3.2 Entity-aware Transformer

As the objects in video are more easily accessible and provide rich indication information for actions, it is natural to narrow down the searching space from all the frames to the actual relevance using the entities in the language query. So we propose an entity-aware Transformer to coarsely select the video clips that are related to the input entity queries. As illustrated in Figure 2, the entity-aware Transformer first learns relationships between video frames via cross-frame attention to provide more contextual information, then fuses the entity query features into each video frame via cross-modal attention, next, attends complementary information across different frames via cross-frame attention, and finally predicts an action-relevant score for each frame to indicate whether the frame is action-relevant or not. According to the predicted action-relevant scores, we select action-relevant video clips where the desired action may happen.



(a) Transformer with frame-wise linear projection for Q,K,V function

(b) Transformer with cross-frame multi-scale LSTM for Q,K,V function

Figure 3: Standard Transformer (a) and our LSTM Transformer (b).

Cross-frame Transformer. For each video V, we extract its visual features $\mathbf{F}_v = [\mathbf{f}_{v,1}, \mathbf{f}_{v,2}, \cdots, \mathbf{f}_{v,T}]^\top \in \mathbb{R}^{T \times d_v}$ by a pre-trained 3D ConvNet, where $\mathbf{f}_{v,i}$ denotes the i-th visual feature with dimension d_v that is computed on a short video clip and T denotes the number of features. A standard Transformer block is then used to attend contextual information across different frames:

$$\mathbf{F}_v^e = Transformer_e(FC_2(\mathbf{F}_v)) \tag{3}$$

where $\mathbf{F}_v^e = [\mathbf{f}_{v,1}^e, \mathbf{f}_{v,2}^e, \cdots, \mathbf{f}_{v,T}^e]^\top \in \mathbb{R}^{T \times d}$ are the updated visual features that pay more attention to the entities; $FC_2(\cdot)$ is a fully connected layer that projects the visual feature from dimension d_v to d; $Transformer_e(\cdot)$ represents a standard Transformer block, as shown in Figure 3(a). We concentrate on the appearance information of frames without considering the temporal information between them, so no position embedding is input to the Transformer.

Cross-modal Fusion. We introduce the context-query attention (CQA) [Zhang *et al.*, 2021a] to integrate the entity query features into visual features of each frame. Given the visual features $\mathbf{F}_v^e$ and the entity query features $\mathbf{F}_q^e$, CQA first computes the their similarity $\mathbf{S} = \mathbf{F}_v^e \cdot \mathbf{F}_q^{e\top} \in \mathbb{R}^{T \times N}$, followed by a row-wise and column-wise softmax normalization to obtain two similarity matrices $\mathbf{S}_r$ and $\mathbf{S}_c$. Then two attention weights are derived by $\mathcal{A}_{VQ} = \mathbf{S}_r \cdot \mathbf{F}_q^e$ and $\mathcal{A}_{QV} = \mathbf{S}_r \cdot \mathbf{S}_c^\top \cdot \mathbf{F}_v^e$. The entity-aware visual features $\mathbf{F}^{ve}$ are computed by

$$\mathbf{F}^{ve} = FC_3([\mathbf{F}_v^e; \mathcal{A}_{VQ}; \mathbf{F}_v^e \odot \mathcal{A}_{VQ}; \mathbf{F}_v^e \odot \mathcal{A}_{QV}]) \tag{4}$$

where $\mathbf{F}^{ve} = [\mathbf{f}_1^{ve}, \mathbf{f}_2^{ve}, \cdots, \mathbf{f}_T^{ve}]^\top \in \mathbb{R}^{T \times d}$; $\odot$ denotes element-wise multiplication; $[\cdot]$ is concatenation; $FC_3(\cdot)$ is

[2]https://www.nltk.org/

a fully connected layer that projects the concatenated feature from dimension $4d$ to d.

Prediction. We calculate the action-relevant score $\mathbf{P}_e = [p_{e,1}, p_{e,2}, \cdots, p_{e,T}]^\top \in \mathbb{R}^T$ of video frames using two fully connected layers for action location prediction:

$$\mathbf{P}_e = sigmoid(FC_5(ReLU(FC_4(\mathbf{F}^{ve})))) \tag{5}$$

where the output feature dimensions of $FC_4(\cdot)$ and $FC_5(\cdot)$ are $\frac{d}{2}$ and 1, respectively. The higher the action-relevant score is, the higher the probability that the corresponding frame is selected as action regions.

3.3 Motion-aware Transformer

Given the coarsely located video clips by the entity-aware Transformer, we propose a motion-aware Transformer to refine the action boundaries by capturing both fine-grained local and global motion changes. As shown in Figure 2, the motion-aware Transformer first learns contextual motion information by a novel LSTM Transformer, then attends the action-relevant parts by the action-relevant score of entity-aware Transformer and fuses motion query features into them via cross-modal attention, next, captures the fine-grained motion changes and global motion changes via the LSTM Transformer, and finally predicts the action boundaries.

LSTM Transformer. The standard Transformer can capture global motion changes due to its capability of modeling long-range dependency where the self-attention module plays a vital role. The self-attention module first conducts linear projections on each input unit to obtain query, key, and value features, and then uses the similarity of query-key feature to aggregate the value features, as shown in Figure 3(a). However, the linear projections cannot capture local motion changes in successive frames. Thus we replace the linear projection with a LSTM cell, as shown in Figure 3(b), which learns sequential local motion changes in videos. In order to deal with the duration variations of the same action in different videos, we apply the long short-term memory at multiple temporal scales.

Specifically, given an input sequence $\mathbf{V}^{in} = \{\mathbf{v}_i^{in}\}_{i=1}^T$, the multi-head self-attention module of our LSTM Transformer is given by $MSA(\mathbf{f}_Q, \mathbf{f}_K, \mathbf{f}_V) = [h_1, h_2, \cdots, h_n]$ where a single head is calculated as $h_i = SA_i(\mathbf{f}_Q, \mathbf{f}_K, \mathbf{f}_V) = softmax(\mathbf{f}_Q \mathbf{f}_K^\top / \sqrt{d})\mathbf{f}_V$, where d is the dimension of intermediate features and $\mathbf{f}_\nu = LSTM_\nu^S(\mathbf{V}^{in}), \nu \in [Q, K, V]$. The $LSTM^S$ is a multi-scale version of the LSTM, denoted as $LSTM^S(\mathbf{V}^{in}) = [L_1(\mathbf{V}^{in}); L_2(\mathbf{V}^{in}); \cdots; L_S(\mathbf{V}^{in})]$, where $[\cdot]$ is concatenation. The s-th scale LSTM is calculated by

$$L_s(\mathbf{V}^{in}) = LSTM^s(\cdots, \mathbf{V}_{i-2s}^{in}, \mathbf{V}_{i-s}^{in}, \mathbf{V}_i^{in}, \mathbf{V}_{i+s}^{in}, \mathbf{V}_{i+2s}^{in}, \cdots). \tag{6}$$

One-time running of L_s can update input sequence every s frames, and sliding L_s in the input sequence one frame for s times can update all input sequence.

For each video V and its visual features $\mathbf{F}_v = [\mathbf{f}_{v,1}, \mathbf{f}_{v,2}, \cdots, \mathbf{f}_{v,T}]^\top \in \mathbb{R}^{T \times d_v}$, the LSTM Transformer is used to learn both fine-grained local and global motion changes:

$$\mathbf{F}_v^m = Transformer_m(FC_2(\mathbf{F}_v)) \tag{7}$$

where $\mathbf{F}_v^m = [\mathbf{f}_{v1}^m, \mathbf{f}_{v2}^m, \cdots, \mathbf{f}_{vN}^m] \in \mathbb{R}^{T \times d}$ are the updated motion features that pay more attention to the motion changes; $FC_2(\cdot)$ is the fully connected layer used in Equation (3) that projects the visual feature from dimension d_v to d; $Transformer_m(\cdot)$ represents the LSTM Transformer.

Cross-modal Fusion. Before cross-modal fusion, we first attend the action-relevant video frames by the action-relevant score $\mathbf{P}_e$ in a soft manner: $\mathbf{F}_v^m = \mathbf{P}_e \odot \mathbf{F}_v^m$, where $\odot$ is an element-wise multiplication. Then motion query features $\mathbf{F}_q^m$ (calculated in Section 3.1) are fused into the visual motion representation $\mathbf{F}_v^m$ via CQA (described in Section 3.2) to obtain the motion-aware visual features:

$$\mathbf{F}^{vm} = CQA(\mathbf{F}_q^m, \mathbf{F}_v^m) \tag{8}$$

where $\mathbf{F}^{vm} = [\mathbf{f}_1^{vm}, \mathbf{f}_2^{vm}, \cdots, \mathbf{f}_T^{vm}]^\top \in \mathbb{R}^{T \times d}$.

Prediction. The start scores $\mathbf{S}_s \in \mathbb{R}^T$ and the end scores $\mathbf{S}_e \in \mathbb{R}^T$ for target action segment are predicted by a two-branch network consisting of two fully connected layers:

$$\begin{aligned} \mathbf{S}_s &= FC_7(ReLU(FC_6(\mathbf{F}^{vm}))) \\ \mathbf{S}_e &= FC_9(ReLU(FC_8(\mathbf{F}^{vm}))) \end{aligned} \tag{9}$$

where the output feature dimensions of $FC_i(\cdot), i \in \{6, 8\}$ and $FC_j(\cdot), j \in \{7, 9\}$ are $\frac{d}{2}$ and 1, respectively. Then the probability distributions of action start and end boundaries are computed by $\mathbf{P}_s^b = softmax(\mathbf{S}_s), \mathbf{P}_e^b = softmax(\mathbf{S}_e) \in \mathbb{R}^T$. Finally, the predicted start and end boundaries of target action segment are derived by maximizing the joint probability:

$$\begin{aligned} (\hat{\tau_s}, \hat{\tau_e}) &= \arg\max_{t_s, t_e} \mathbf{P}_s^b(t_s) \times \mathbf{P}_e^b(t_e), \\ p_{se}^b &= \mathbf{P}_s^b(\hat{\tau_s}) \times \mathbf{P}_e^b(\hat{\tau_e}) \end{aligned} \tag{10}$$

where p_{se}^b is the optimized score of the predicted boundaries $(\hat{\tau_s}, \hat{\tau_e})$. We also apply another branch of two fully connected layers network to predict a inner probability for each frame as a auxiliary task only for training [Wang *et al.*, 2021]. Let $\mathbf{P}^{in} = [\mathbf{p}_1^{in}, \mathbf{p}_2^{in}, \cdots, \mathbf{p}_T^{in}]^\top \in \mathbb{R}^T$ denote the probability of being action frames, calculated by

$$\mathbf{P}^{in} = sigmoid(FC_b(ReLU(FC_a(\mathbf{F}^{vm})))) \tag{11}$$

where the output feature dimensions of $FC_a(\cdot)$ and $FC_b(\cdot)$ are $\frac{d}{2}$ and 1, respectively.

3.4 Training Objective

Given the predicted probability distribution of action boundaries $\mathbf{P}_s^b$ and $\mathbf{P}_e^b$, the training objective for action boundary prediction is formulated by

$$\mathcal{L}^{boundary} = f_{XE}(\mathbf{P}_s^b, \tau_s) + f_{XE}(\mathbf{P}_e^b, \tau_e) \tag{12}$$

where $f_{XE}(\cdot)$ is a cross-entropy function, and (τ_s, τ_e) are the ground-truth boundaries. Given the inner probability $\mathbf{P}^{in}$, the training objective for action frame prediction is formulated by

$$\mathcal{L}^{inner} = f_{BXE}(\mathbf{P}^{in}, \mathbf{Y}^{in}) \tag{13}$$

where $f_{BXE}(\cdot)$ is a binary cross-entropy function and $\mathbf{Y}^{in} = \{y_i^{in}\}_{i=1}^T \in \{0, 1\}$, when $\tau_s \le i \le \tau_e$, $y_i^{in} = 1$, otherwise $y_i^{in} = 0$. The overall objective is given by

$$\mathcal{L} = \lambda_1 \mathcal{L}^{boundary} + \lambda_2 \mathcal{L}^{inner} \tag{14}$$

where λ_1 and λ_2 are hyper-parameters.

Methods	$R@1; IoU = \mu$ 0.3	0.5	0.7	$mIoU$
VSL [Zhang *et al.*, 2021a]	70.46	54.19	35.22	50.02
LGI [Mun *et al.*, 2020]	72.96	59.46	35.48	51.38
DeNet [Zhou *et al.*, 2021]	-	59.7	38.52	-
SS [Ding *et al.*, 2021]	-	60.75	36.19	-
CPNet [Li *et al.*, 2021]	71.94	60.27	38.74	52.00
ACRM [Tang *et al.*, 2022]	73.47	57.53	38.33	-
ICG [Nan *et al.*, 2021]	67.63	50.24	32.88	48.02
CPN [Zhao *et al.*, 2021]	68.48	51.07	31.54	48.08
SPA [Zhang *et al.*, 2021b]	73.84	60.86	41.34	53.92
CBLN [Liu *et al.*, 2021]	-	61.13	38.22	-
Ours	**74.19**	**61.69**	**41.96**	**54.45**

Table 1: Comparison with the state-of-the-art methods on the Charades-STA dataset.

Methods	$R@1; IoU = \mu$ 0.3	0.5	0.7	$mIoU$
BPNet [Xiao *et al.*, 2021]	25.96	20.96	14.08	19.53
VSL [Zhang *et al.*, 2021a]	29.61	24.27	20.03	24.11
I^2N [Ning *et al.*, 2021]	31.80	28.69	-	-
SS [Ding *et al.*, 2021]	41.33	29.56	-	-
CPNet [Li *et al.*, 2021]	42.61	28.29	-	28.69
CBLN [Liu *et al.*, 2021]	38.89	27.65	-	-
ICG [Nan *et al.*, 2021]	38.84	29.07	19.05	28.26
SMIN [Wang *et al.*, 2021]	48.01	35.24	-	-
CPN [Zhao *et al.*, 2021]	48.29	36.58	21.25	34.63
Ours	**50.11**	**38.16**	**26.82**	**36.43**

Table 2: Comparison with the state-of-the-art methods on the TACoS dataset.

Methods	$R@1; IoU = \mu$ 0.3	0.5	0.7	$mIoU$
Ours w/o EA Trans	71.15	58.25	38.79	51.87
FC Trans	67.18	48.14	27.69	46.24
T-Conv Trans	73.60	55.86	37.39	53.20
T-Conv	61.64	37.34	21.91	42.50
LSTM	72.34	59.14	40.53	53.12
Ours	**74.19**	**61.69**	**41.96**	**54.45**

Table 3: Ablation studies on the Charades-STA dataset.

4 Experiments

4.1 Datasets and Evaluation Metrics

We evaluate our method on two datasets: **Charades-STA** [Gao *et al.*, 2017] and **TACoS** [Regneri *et al.*, 2013]. The Charades-STA dataset is built on the Charades dataset [Sigurdsson *et al.*, 2016] and contains 16,128 annotations, including 12,408 for training and 3,720 for test. The TACoS dataset is built on the MPII Cooking Compositive dataset [Rohrbach *et al.*, 2012] and contains 18,818 annotations, including 10146 for training, 4589 for validation, and 4083 for test.

We use the metrics of $R@n; IoU = \mu$ and $mIoU$ for evaluation. $R@n; IoU = \mu$ denotes the percentage of test samples that have at least one result whose IoU with ground-truth is larger than μ in top-n predictions and $mIoU$ denotes the average IoU over all test samples. We set $n = 1$ and $\mu \in [0.3, 0.5, 0.7]$.

4.2 Implementation Details

Following the previous methods, 3D convolutional features (C3D for TACoS, and I3D for Charades-STA) are extracted to encode videos. We adopt Adam [Kingma and Ba, 2014] for optimization with an initial learning rate of 5e-4 and a linear decay schedule. The loss weights λ_1 and λ_2 in Equation (14) are set to 1 and 10, respectively. The number of Transformer Blocks is set to 1 and 3 for early and late Transformers in entity-aware and motion-aware Transformers. The feature dimension of all intermediate layers is set to 512, the head number of multi-head self-attention is set to 8, the layer number and scale number of long short-term memory are set to 1 and 3, respectively.

4.3 Comparison Results

We compare our method with the latest state-of-the-art methods on the Charades-STA and TACoS datasets in Table 1 and Table 2, respectively. From the results, it is interesting to observe that our method achieves the best performance in terms of all evaluation metrics on both two datasets, clearly validating the superiority of the proposed entity-aware and motion-aware Transformers on improving the localization precision via a coarse-to-fine strategy.

4.4 Ablation Studies

We perform in-depth ablation studies to evaluate each component of our method on the Charades-STA dataset. The results are shown in Table 3.

Effect of Entity-aware Transformer. To evaluate the entity-aware Transformer, we design a baseline model called "Ours w/o EA Trans" that uses only the motion-aware Transformer with the input word feature of the language query. As shown in Table 3, our method outperforms "Ours w/o EA Trans" with gains of 3% on all evaluation metrics, clearly demonstrating the effectiveness of the entity-aware Transformer.

Analysis of Motion-aware Transformer. To evaluate the Motion-aware Transformer, we design several variants of our method for comparison, denoted as "FC Trans", "T-Conv Trans", "T-Conv" and "LSTM": (i) "FC Trans" and "T-Conv Trans" replace the LSTM cell by fully connected layers and temporal convolutional layers, respectively. So "FC Trans" degrades into a standard Transformer; (ii) "T-Conv" and "LSTM" replace the LSTM Transformer by temporal convolutional and LSTM layers, respectively. For a fair comparison, "T-Conv Trans" and "T-Conv" have multiple kernel sizes of 3, 5, and 7, and "LSTM" has the same multi-scale LSTM as LSTM Transformer. From the result in Table 3, we have the following observations. (1) Compared with "T-Conv Trans", our method achieves better results with gains of 5.83% on $R@1; IoU = 0.5$ and 4.57% on $R@1; IoU = 0.7$. Moreover, "T-Conv Trans" outperforms "FC Trans" by 7.72% on $R@1; IoU = 0.5$ and 9.70% on $R@1; IoU = 0.7$. These validate that carefully modeling of local motion changes significantly improves the localization accuracy. (2) Compared with "LSTM", our method achieves better results. Moreover, "T-Conv Trans" outperforms "T-

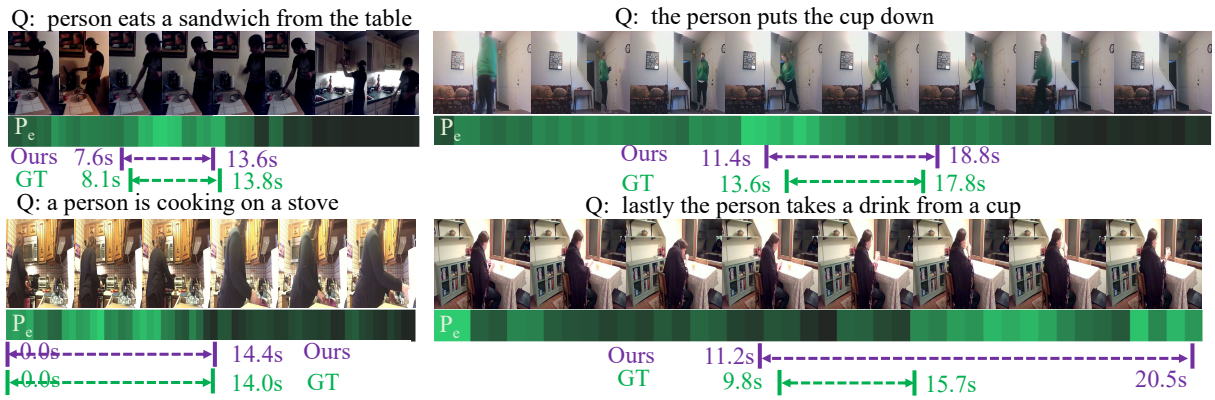


Figure 4: Examples of action localization visualization on the Charades-STA dataset. The action-relevant score P_e is predicted by the entity-aware Transformer and brighter colors indicate higher values.

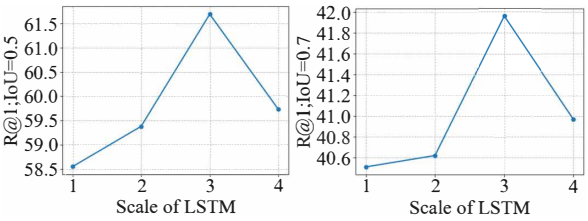


Figure 5: Analysis of the effect of scale number in LSTM Transformer on the Charades-STA dataset.

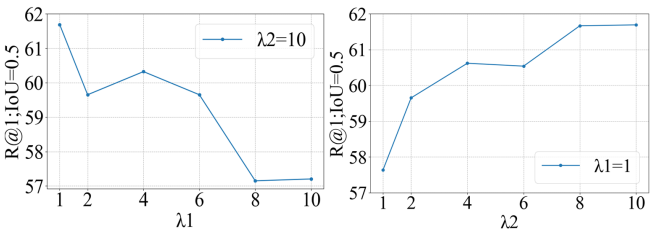


Figure 6: Analysis of the effect of loss weights on the Charades-STA dataset.

Conv" by more than 10% on all evaluation metrics. These show that the global changes captured by Transformer are beneficial to action localization.

4.5 Parameter Analysis

Temporal Scale Number in LSTM Transformer. The performances of different temporal scales in LSTM Transformer on the Charades-STA dataset are shown in Figure 5. We observe that when the scale number increases, the performance first increases and then gradually decreases, which demonstrates that modeling local motion changes at more temporal scales can improve the location precision, but also may bring redundant information.

Loss Weights. To analyze the effect of the loss weights λ_1 and λ_2 in Equation (14), we vary the value of λ_1 in [1,10] and the value of λ_2 in [1,10]. The results are shown in Figure 6. It is interesting to observe that when λ_1 increases, the performance drops dramatically. In contrast, the performance improves along with the increasing λ_2, which shows that larger λ_2 boosts the per-frame inner prediction, and thus helps the boundary localization.

4.6 Qualitative Analysis.

We show several examples of action localization results on the Charades-STA dataset in Figure 4 by visualizing the corresponding action-relevant scores predicted by the entity-aware Transformer where bright colors indicate higher values. From the first three cases, we see that the action-relevant scores make it easier to accurately localize the action boundaries by paying more attention to the target action in a shrunken temporal. However, in the last case, the entities ("*person & cup*") remain unchanged that the action-relevant scores of different frames are similar (the margin between maximum and minimum is less than 0.1), thus contributing less to the final boundary localization. Moreover, the entity word "*drink*" is wrongly classified to a motion word, so that the predicted boundaries wrongly fall in the boundaries of "*drink*".

5 Conclusion

We have presented a novel coarse-to-fine model called entity-aware and motion-aware Transformers for language-driven action localization. It can progressively predict the action boundaries with high precision by first attending the action-relevant clips via the entity-aware Transformer and then refining the start and end frames via the motion-aware Transformer. By integrating multi-scale long short-term memory cells into the self-attention module, the motion-aware Transformer succeeds in capturing the fine-grained motion changes, thus achieving promising results. Extensive experiments on two public datasets have demonstrated that our method outperforms the state-of-the-art methods. In the future work, we are going to apply the entity-aware and motion-aware Transformers to weakly-supervised language-driven action localization.

Acknowledgments

This work was supported in part by the Natural Science Foundation of China (NSFC) under Grant No 62072041.

References

[Chen and Jiang, 2019] Shaoxiang Chen and Yu-Gang Jiang. Semantic proposal for activity localization in videos via sentence query. In *Proc. of AAAI*, 2019.

[Ding *et al.*, 2021] Xinpeng Ding, Nannan Wang, et al. Support-set based cross-supervision for video grounding. In *Proc. of ICCV*, 2021.

[Gao *et al.*, 2017] Jiyang Gao, Chen Sun, et al. Tall: Temporal activity localization via language query. In *Proc. of ICCV*, 2017.

[Ghosh *et al.*, 2019] Soham Ghosh, Anuva Agarwal, et al. Excl: Extractive clip localization using natural language descriptions. In *Proc. of ACL*, 2019.

[Hahn *et al.*, 2019] Meera Hahn, Asim Kadav, et al. Tripping through time: Efficient localization of activities in videos. In *BMVC*, 2019.

[Hendricks *et al.*, 2017] Lisa Anne Hendricks, Oliver Wang, et al. Localizing moments in video with natural language. In *Proc. of ICCV*, 2017.

[Kingma and Ba, 2014] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[Li *et al.*, 2021] Kun Li, Dan Guo, and Meng Wang. Proposal-free video grounding with contextual pyramid network. In *Proc. of AAAI*, 2021.

[Liu *et al.*, 2018] Meng Liu, Xiang Wang, et al. Attentive moment retrieval in videos. In *Proc. of ACM SIGIR*, 2018.

[Liu *et al.*, 2020] Daizong Liu, Xiaoye Qu, et al. Jointly cross- and self-modal graph attention network for query-based moment localization. In *Proc. of ACM MM*, 2020.

[Liu *et al.*, 2021] Daizong Liu, Xiaoye Qu, et al. Context-aware biaffine localizing network for temporal sentence grounding. In *Proc. of CVPR*, 2021.

[Lu *et al.*, 2019] Chujie Lu, Long Chen, et al. Debug: A dense bottom-up grounding approach for natural language video localization. In *Proc. of EMNLP*, 2019.

[Mun *et al.*, 2020] Jonghwan Mun, Minsu Cho, and Bohyung Han. Local-global video-text interactions for temporal grounding. In *Proc. of CVPR*, 2020.

[Nan *et al.*, 2021] Guoshun Nan, Rui Qiao, Yao Xiao, et al. Interventional video grounding with dual contrastive learning. In *Proc. of CVPR*, 2021.

[Ning *et al.*, 2021] Ke Ning, Lingxi Xie, et al. Interaction-integrated network for natural language moment localization. *TIP*, 2021.

[Pennington *et al.*, 2014] Jeffrey Pennington, Richard Socher, et al. Glove: Global vectors for word representation. In *Proc. of EMNLP*, 2014.

[Qu *et al.*, 2020] Xiaoye Qu, Pengwei Tang, et al. Fine-grained iterative attention network for temporal language localization in videos. In *Proc. of ACM MM*, 2020.

[Regneri *et al.*, 2013] Michaela Regneri, Marcus Rohrbach, et al. Grounding action descriptions in videos. *TACL*, 2013.

[Rohrbach *et al.*, 2012] Marcus Rohrbach, Michaela Regneri, et al. Script data for attribute-based recognition of composite activities. In *ECCV*, 2012.

[Sigurdsson *et al.*, 2016] Gunnar A Sigurdsson, Gül Varol, et al. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *Proc. of ECCV*, 2016.

[Tang *et al.*, 2022] Haoyu Tang, Jihua Zhu, et al. Frame-wise cross-modal matching for video moment retrieval. *IEEE Transactions on Multimedia*, 2022.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, et al. Attention is all you need. In *NeurIPS*, 2017.

[Wang *et al.*, 2021] Hao Wang, Zheng-Jun Zha, et al. Structured multi-level interaction network for video moment localization via language query. In *Proc. of CVPR*, 2021.

[Wu *et al.*, 2020] Jie Wu, Guanbin Li, et al. Tree-structured policy based progressive reinforcement learning for temporally language grounding in video. In *Proc. of AAAI*, 2020.

[Xiao *et al.*, 2021] Shaoning Xiao, Long Chen, et al. Boundary proposal network for two-stage natural language video localization. In *Proc. of AAAI*, 2021.

[Yuan *et al.*, 2019] Yitian Yuan, Tao Mei, and Wenwu Zhu. To find where you talk: Temporal sentence localization in video with attention based location regression. In *Proc. of AAAI*, 2019.

[Yuan *et al.*, 2020] Yitian Yuan, Lin Ma, et al. Semantic conditioned dynamic modulation for temporal sentence grounding in videos. *TPAMI*, 2020.

[Zeng *et al.*, 2020] Runhao Zeng, Haoming Xu, et al. Dense regression network for video grounding. In *Proc. of CVPR*, 2020.

[Zhang *et al.*, 2020] Songyang Zhang, Houwen Peng, and Jiebo Luo. Learning 2d temporal adjacent networks for moment localization with natural language. In *Proc. of AAAI*, 2020.

[Zhang *et al.*, 2021a] Hao Zhang, Aixin Sun, et al. Natural language video localization: A revisit in span-based question answering framework. *TPAMI*, 2021.

[Zhang *et al.*, 2021b] Hao Zhang, Aixin Sun, Wei Jing, et al. Parallel attention network with sequence matching for video grounding. In *Proc. of ACL*, 2021.

[Zhang *et al.*, 2021c] Mingxing Zhang, Yang Yang, et al. Multi-stage aggregated transformer network for temporal language localization in videos. In *Proc. of CVPR*, 2021.

[Zhao *et al.*, 2021] Yang Zhao, Zhou Zhao, et al. Cascaded prediction network via segment tree for temporal video grounding. In *Proc. of CVPR*, 2021.

[Zhou *et al.*, 2021] Hao Zhou, Chongyang Zhang, et al. Embracing uncertainty: Decoupling and de-bias for robust temporal grounding. In *Proc. of CVPR*, 2021.