Dynamic Pathway for Query-aware Feature Learning in Language-driven Action Localization

Shuo Yang, Xinxiao Wu, *Member, IEEE*, Zirui Shang, Jiebo Luo, *Fellow, IEEE*

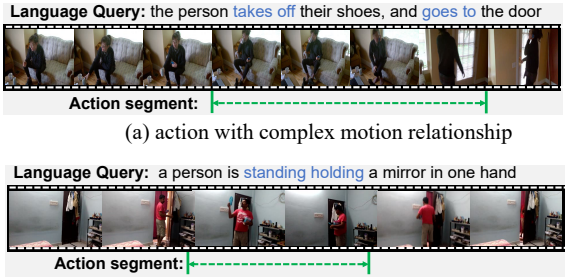
***Abstract*—Language-driven action localization aims to search a video segment in an untrimmed video, which is semantically relevant to an input language query. This task is challenging since language queries describe diverse actions with different motion characteristics and semantic granularities. Some actions, such as *"the person takes off their shoes, and goes to the door"*, are characterized by complex motion relationships, while others, such as *"a person is standing holding a mirror in one hand"*, are distinguished by salient body postures. In this paper, we propose a dynamic pathway between an exploitation module and an exploration module for query-aware feature learning to handle the diversity of actions. The exploitation module works in a coarse-to-fine manner, first learns the feature of general motion relationships to search the coarse segment of the target action and then learns the feature of subtle motion changes to predict the refined action boundaries. The exploration module functions in a point-to-area diffusion fashion, first learns the feature of sub-action pattern to search the salient postures of the target action and then learns the feature of temporal dependency to expand the posture frames to the action segment. The exploitation module and the exploration module are dynamically and adaptively selected to learn comprehensive representations of diverse actions to improve the action localization accuracy. Extensive experiments on the Charades-STA and TACoS datasets demonstrate that our method performs better than existing methods.**

***Index Terms*—Dynamic pathway, exploitation, exploration, language-driven action localization, video grounding, video moment retrieval.**

I. Introduction

With the explosive growth in the number of videos on the internet, searching content-of-interest videos draws growing attention from both industry and academia. In this paper, we focus on the task of language-driven action localization, also known as video moment retrieval or video grounding, which aims to search a video segment in an untrimmed video that is semantically relevant to an input language query. It is an important task in video understanding and has wide applications in robotic navigation and human-computer interaction.

To address this task, numerous methods focus on cross-modality alignment between videos and language queries to bridge the huge cross-modal gap between visual and textual features, such as co-attention [1], cross-modal graph attention [2], context-query attention [3], and cross-attention [4]. These methods have achieved promising results in recent years. However, they use a static network to deal with different kinds of actions, while neglecting various motion characteristics and semantic granularities of actions described by language queries.



(a) action with complex motion relationship

(b) action with salient body posture

Fig. 1. Examples of different kinds of actions described by language queries. (a) describes an action with complex motion relationship. (b) describes an action with salient body posture.

Different from the action-related close-set tasks, such as action recognition or action localization, with pre-defined action categories, the language-driven action localization task describes diverse actions that demonstrate various motion relationships and salient body postures at different semantic granularities. For example, as shown in Figure 1(a), the action *"the person takes off their shoes, and goes to the door"* involves several sequential motions (sub-actions) *"take off shoes"*, and *"go to the door"*, and its feature need to be learned to capture temporal contextual relationships between motions. On the other hand, as shown in Figure 1(b), the action *"a person is standing holding a mirror in one hand"* can be easily located by the salient posture *"a standing person is holding a mirror"*, whose feature need to be learned to distinguish different postures. Therefore, it is extremely challenging to learn adaptive query-aware features to represent different kinds of actions in the task of language-driven action localization.

To address this challenge, we propose a dynamic pathway between an exploitation module and an exploration module for query-aware feature learning, which can adaptively select an appropriate feature learning pathway according to a specific language query. The exploitation module works in a coarse-to-fine manner to handle the actions with coarse semantic granularity and complex motion changes. It first learns the

Shuo Yang and Xinxiao Wu are with the Beijing Laboratory of Intelligent Information Technology, School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China, and also with the Guangdong Provincial Laboratory of Machine Perception and Intelligent Computing, Shenzhen MSU-BIT University, Shenzhen 518172, China (e-mail: shuoyang@bit.edu.cn, wuxinxiao@bit.edu.cn). Xinxiao Wu is the corresponding author;

Zirui Shang are with the Beijing Key Laboratory of Intelligent Information Technology, School of Computer Science and Technology, Beijing Institute of Technology, China (e-mail: shangzirui@bit.edu.cn);

Jiebo Luo is with the Department of Computer Science, University of Rochester, Rochester, NY 14627 USA (e-mail: jluo@cs.rochester.edu).

feature of general motion relationships to search the coarse segment of the target action and then learns that of subtle motion changes to predict the refined action boundaries. Starting with the features of general motion relationships, most video segments irrelevant to the input query are then excluded, thus making the localization of refined action boundaries easier and more precise. The exploration module functions in a point-to-area diffusion fashion to handle the actions with fine semantic granularity and salient body postures. It first learns the feature of the sub-action pattern to search the salient postures of the target action and then learns that of temporal dependency to expand the posture frames to the action segment. Starting with the feature of the sub-action pattern, salient postures are then be detected more easily, which gives a strong indicator for the action segment and can be used as anchors for expanding to the action segment boundaries. The exploitation module and exploration module work in a mutually complementary manner. By adding a query-aware adaptive module between them, an input can be adaptively processed, thus contributing to a flexible, accurate search of the target action segments. Furthermore, we design a query-aware selection module to dynamically and flexibly select the exploitation module or the exploration module to adaptively learn the action feature according to the input language query, thereby improving the accuracy and interpretability of searching for the target action.

Specifically, the exploitation module consists of a multi-scale long short-term memory (LSTM) block and a self-attention block in parallel, followed by a multi-scale temporal convolution block. The exploration module consists of a multi-scale temporal convolution block, followed by a multi-scale LSTM block and a self-attention block in parallel. The architecture of the blocks in the exploitation module is the same as that of the exploration module, but for different purposes. Note that our LSTM block and temporal convolution block are both implemented in multiple temporal scales to deal with the various duration of the same action in different videos and the different granularities of motion relationships. The query-aware module predicts a two-dimensional one-hot vector based on the language representation, which is a weighted sum of features of all the input language tokens. Extensive experiments on the Charades-STA and TACoS datasets demonstrate that our method outperforms the existing methods.

Our main contributions are summarized as follows:

- We propose a dynamic pathway between multiple modules to handle diverse actions with different motion characteristics and semantic granularities described by language queries. To the best of our knowledge, this is the first attempt at leveraging a dynamic network structure for different actions in language-driven action localization.
- We design an exploitation module to deal with actions with coarse semantic granularities and complex motion changes of multiple sub-actions, which localizes the target action in a coarse-to-fine manner.
- We design an exploration module to deal with actions with fine semantic granularities and salient body postures, which localizes the target segment in a point-to-area diffusion fashion.

II. Related Work

The language-driven action localization task was first proposed in [5], [6] and further studied by researchers using mainly two lines of methods: proposal-based [2], [5]–[17] and proposal-free [18]–[22].

A. Proposal-based Methods

The proposal-based methods first generate proposals using temporal bounding boxes (*e.g.,* sliding windows [5] and a 2D temporal adjacency map [11]), then calculate the similarities between the proposals and the given language query, and finally rank all the proposals by the similarities. MCN [6] learns a joint space of queries and proposals for better similarity measurement between visual and language representations. MHST [23] generates proposals in a tree structure, which merges the adjacent frames sharing the same visual-linguistic semantics into the parent node. To enhance the proposal feature, 2D-TAN [11] learns temporal relations between adjacent video moments using a 2D temporal proposal map, MSAT [24] introduces a multi-stage aggregated transformer that uses a BERT-variant transformer backbone to extract visual-language features, SCDM [10] modulates the temporal convolutional visual features to correlate and compose language-related video contents using a semantic conditioned dynamic modulation algorithm, and SLP [25] adopts a two-step human-like framework to take both frame-differentiable and boundary-precise requirements into account.

B. Proposal-free Methods

The proposal-based methods are time-consuming and often easily introduce redundant candidates. Consequently, more efficient proposal-free methods are proposed as alternatives, which directly predict the temporal location of target action by fusing the visual and linguistic features. ExCL [26] and SeqPAN [27] model the cross-modal interaction between the language and video to predict the start and end time of the target action. VSLNet [3] searches the target action within a highlighted region in a span-based question-answering framework. To take advantage of the structural information of videos and queries, several other methods have been proposed. LGI [28] uses a sequential query attention module to extract the implicit semantic information from local to global. CPNet [29] proposes a pyramid network to extract 2D contextual correlation maps at different temporal scales, which progressively replenishes the temporal contexts and refines the location of the target action by enlarging the temporal receptive fields. MGPN [30] perceives intra-modality and inter-modality information at a multi-granularity level, leveraging fine-grained intra-modality clues to explore deeper inter-modality information.

Both the proposal-based methods and the proposal-free methods perform inference in a static manner, that is to say, the computational graph is fixed once trained and is thus unable to represent various motion characteristics and semantic granularities of actions, limiting their representation capabilities. In this paper, we attempt to handle different actions

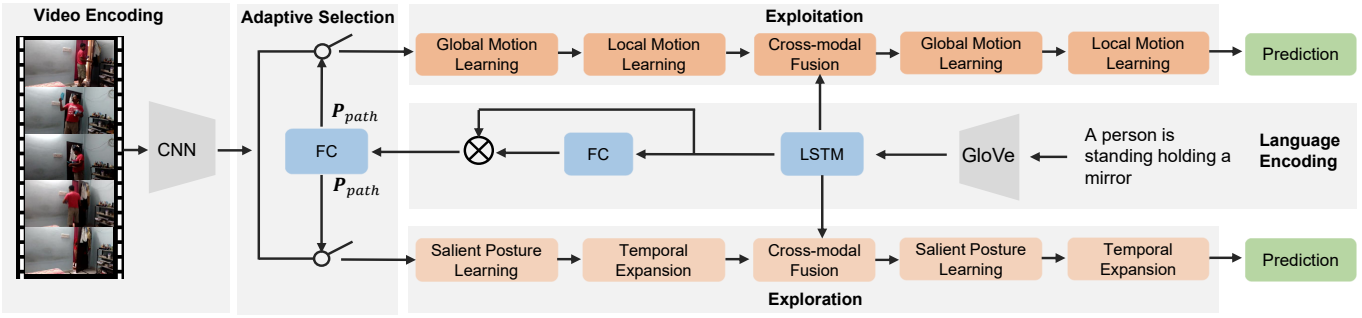


Fig. 2. Overview of the proposed dynamic pathway between an *exploitation* module and an *exploration* module. $\mathbf{P}_{path}$ decides which path, the exploitation module or the exploration module, is selected to learn video features according to the sentence-level feature of language queries.

by proposing a dynamic pathway where multiple modules are flexibly and dynamically selected to learn adaptive action features based on the semantics of language queries. This is the first attempt at designing a dynamic architecture for language-driven action localization and making a precise estimation of action boundaries.

III. Our Method

Language-driven action localization aims to search a target action segment (τ_s, τ_e) corresponding to the language query $S = \{w_i\}_{i=1}^{N_q}$ from an untrimmed video $V = \{v_t\}_{t=1}^{N_v}$, where τ_s and τ_e represent the start frame and end frame of the action segment, respectively, w_i represents the i-th word in the query, v_t represents the t-th video frame, and N_v and N_q represent the numbers of video frames and text words, respectively.

Our method consists of five modules: a feature encoding module, an exploitation module, an exploration module, a query-aware adaptive selection module, and a prediction module. The feature encoding module encodes the input language query and video. The exploitation module first learns the feature of general motion relationships to search the coarse segment of the target action, and then learns that of subtle motion changes to predict the refined action boundaries. The exploration module first learns the feature of the sub-action pattern to search the salient postures of the target action and then learns that of temporal dependency to expand the posture frames to the action segment. The query-aware adaptive selection module dynamically and flexibly selects the exploitation module or the exploration module to learn the action feature adaptively according to the input language query. The prediction module predicts the probabilities for each frame being the boundary of the target action segment. Figure 2 shows the overview of our method.

A. Feature Encoding Module

1) Language Encoding: Given an input language query S, its word features $\mathbf{Q} = [\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_{N_q}]^\top \in \mathbb{R}^{N_q \times d_w}$ are first initialized using the GloVe embedding [31], where $\mathbf{w}_i$ denotes the i-th word feature with dimension d_w, and N_q denotes the number of words in S. Then a three-layer bi-directional LSTM is used to learn the relationships of words, followed by a feed-forward network (FFN). Finally, the token weights $\boldsymbol{\alpha} \in \mathbb{R}^{N_q}$ of words are learned through two fully connected (FC) layers, and a sentence-level query feature is calculated by a weighted sum of all the words. The overall language encoding is formulated by

$$\begin{aligned} \mathbf{F}_q &= FFN_1(LSTM(\mathbf{Q})) \\ \boldsymbol{\alpha} &= softmax(FC_2(\delta(FC_1(\mathbf{F}_q)))) \\ \mathbf{F}_s &= \boldsymbol{\alpha}^\top \cdot \mathbf{F}_q \end{aligned} \tag{1}$$

where $\mathbf{F}_q = [\mathbf{f}_{q,1}, \mathbf{f}_{q,2}, \cdots, \mathbf{f}_{q,N_q}]^\top \in \mathbb{R}^{N_q \times d}$ are the encoded linguistic query features with dimension d; $\mathbf{F}_s \in \mathbb{R}^d$ is the sentence-level query feature; $FFN_1(\cdot)$ is the feed-forward network that consists of a linear layer and a $ReLU$ layer; $FC_1(\cdot)$ and $FC_2(\cdot)$ are fully connected layers, and their output dimensions are $\frac{d}{2}$ and 1, respectively, and the δ is $ReLU$ activation layer.

2) Video Encoding: Each input video V is divided into a sequence of non-overlap clips with a fixed length (*e.g.*, 16 frames). Then the visual feature of each clip is extracted using a pre-trained 3D-CNN [32], [33]. Finally, the clip features are fed into a feed-forward network to have the same dimension as the query features. The overall video encoding is formulated by

$$\mathbf{F}_v = FFN_2(3D\text{-}CNN(V)) \tag{2}$$

where $\mathbf{F}_v = [\mathbf{f}_{v,1}, \mathbf{f}_{v,2}, \cdots, \mathbf{f}_{v,T}]^\top \in \mathbb{R}^{T \times d}$ are the encoded video features with dimension d and T is the number of video clips; $FFN_2(\cdot)$ is the feed-forward network that consists of a linear layer and a $ReLU$ layer.

B. Exploitation Module

Some language queries describe actions with various motion relationships, such as *"the person takes off their shoes, and goes to the door"* and *"the woman mixed all ingredients, put it in a pan and put it in the oven"*. Features of these actions should be learned to capture temporal contextual relationships between motions. To that end, we design an exploitation module, which consists of a global motion learning neural network and a local motion learning neural network. The global motion learning network learns general motion relationships to roughly localize the target action area, and the local motion learning network learns subtle motion changes to refinely estimate the action boundaries.

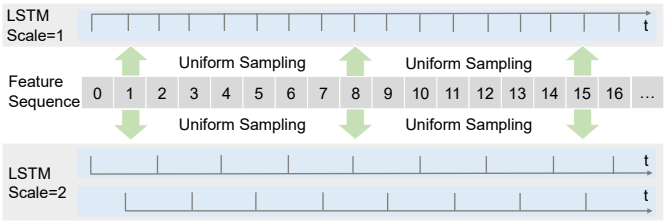


Fig. 3. An example of multi-scale LSTM with two temporal scales.

1) Global Motion Learning Network: The global motion learning network consists of a multi-scale LSTM block and a multi-head self-attention block, which stack in parallel. The LSTM block exploits the sequential nature of videos to learn temporal motion relationships between consecutive frames in a global manner. Since the same action may last for different durations in different videos due to the diversity of subjects and motion styles, we implement the LSTM block in a multi-scale version, denoted by $MS\text{-}LSTM$, as shown in Figure 3.

Given input video features $\mathbf{F}_v$, the multi-scale LSTM is formulated by

$$MS\text{-}LSTM(\mathbf{F}_v) = FFN_3([L_1(\mathbf{F}_{v_{in}}); L_2(\mathbf{F}_{v_{in}}); \cdots; L_S(\mathbf{F}_{v_{in}})]) \tag{3}$$

where $[\cdot]$ is a concatenation operation; $FFN_3(\cdot)$ is a feed-forward network; $\mathbf{F}_{v_{in}}$ are subset video features uniform sampled from $\mathbf{F}_v$; $L_s(\mathbf{F}_{v_{in}})$ represents the $s\text{-}th$ scale LSTM, $s \in \{1, 2, \cdots, S\}$, formulated by

$$L_s(\mathbf{F}_{v_{in}}) = \begin{cases} LSTM_s(\mathbf{f}_{v,0}, \mathbf{f}_{v,k}, \mathbf{f}_{v,2k}, \mathbf{f}_{v,3k}, \cdots) \\ LSTM_s(\mathbf{f}_{v,1}, \mathbf{f}_{v,k+1}, \mathbf{f}_{v,2k+1}, \mathbf{f}_{v,3k+1}, \cdots) \\ \cdots \\ LSTM_s(\mathbf{f}_{v,k-1}, \mathbf{f}_{v,2k-1}, \mathbf{f}_{v,3k-1}, \mathbf{f}_{v,4k-1}, \cdots) \end{cases} \tag{4}$$

Specifically, for the $s\text{-}th$ scale LSTM, the video features $\mathbf{F}_v \in \mathbb{R}^{T\times d}$ are split into s subsets, and the parameters of $LSTM_s$ are shared for these subsets. And for the i-th subset, starting from frame i-1, features are sampled every s step from the input T video features.

To simultaneously model the non-local motion relationships, We also conduct a multi-head self-attention block to capture the long-distance dependencies between video features. For the input video features $\mathbf{F}_v$, the multi-head self-attention is first used, denoted by $MSA(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = [h_1, h_2, \cdots, h_n]$, where each single head is calculated as $h_i = SA_i(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = softmax(\mathbf{Q}\mathbf{K}^\top/\sqrt{d})\mathbf{V}$. Then a residual connection, a layer normalization (LN), and a feed-forward network are used. So the overall multi-head self-attention is formulated by

$$\begin{aligned} &MSAB(\mathbf{F}_v) = FFN_4(LN(\mathbf{H}_v) + \mathbf{F}_v)), \\ &\mathbf{H}_v = MSA(FC_Q(\mathbf{F}_v), FC_K(\mathbf{F}_v), FC_V(\mathbf{F}_v)) \end{aligned} \tag{5}$$

where $FC_j(\cdot)$ denotes fully connected layer, $j \in \{Q, K, V\}$.

Finally, the video features $\mathbf{F}_v^g = [\mathbf{f}_{v,1}^g, \mathbf{f}_{v,2}^g, \cdots, \mathbf{f}_{v,T}^g]^\top \in \mathbb{R}^{T\times d}$ learned through the global motion learning network are given by

$$\mathbf{F}_v^g = MS\text{-}LSTM(\mathbf{F}_v) + MSAB(\mathbf{F}_v). \tag{6}$$

2) Local Motion Learning Network: The local motion learning network consists of a multi-scale temporal convolutional block to learn subtle local motion changes at multiple temporal scales to further identify the action boundaries. Given input video features $\mathbf{F}_v^g$, K temporal convolution layers $G_k(\cdot)_{k=1}^K$ with different kernels are applied first and followed by $ReLU$ activation layers, and then the outputs are concatenated and fed into a feed-forward network to maintain the feature dimension as d. The overall local motion learning network is summarized as

$$\mathbf{F}_v^c = FFN_5([\delta(G_1(\mathbf{F}_v^g)), \delta(G_2(\mathbf{F}_v^g)), \cdots, \delta(G_K(\mathbf{F}_v^g))]), \tag{7}$$

where $\mathbf{F}_v^c = [\mathbf{f}_{v,1}^c, \mathbf{f}_{v,2}^c, \cdots, \mathbf{f}_{v,T}^c]^\top \in \mathbb{R}^{T\times d}$ represent the video features learned through the local motion learning network; $[\cdot]$ is concatenation; $FFN_5(\cdot)$ is a feed-forward network that projects the concatenated feature from dimension Kd to d, and the δ is $ReLU$ activation layer.

3) Cross-modal Fusion Network: We design a cross-modal fusion network to integrate the language query features into the video features via context-query attention [3], [34]. Given the video features $\mathbf{F}_v^c$ and the language query features $\mathbf{F}_q$, their similarities $\mathbf{Sim} = SIM(\mathbf{F}_v^c, \mathbf{F}_q) \in \mathbb{R}^{T\times N_q}$ are computed first, followed by a row-wise and a column-wise softmax normalization to obtain two similarity matrices $\mathbf{S}_r$ and $\mathbf{S}_c$. And then two attention weights are derived by $\mathcal{A}_{VQ} = \mathbf{S}_r \cdot \mathbf{F}_q$ and $\mathcal{A}_{QV} = \mathbf{S}_r \cdot \mathbf{S}_c^\top \cdot \mathbf{F}_v^c$. The fused visual-linguistic features $\mathbf{F}_{vq}$ are computed by

$$\mathbf{F}_{vq}^{e1} = FFN_6([\mathbf{F}_v^c; \mathcal{A}_{VQ}; \mathbf{F}_v^c \odot \mathcal{A}_{VQ}; \mathbf{F}_v^c \odot \mathcal{A}_{QV}]) \tag{8}$$

where $\mathbf{F}_{vq}^{e1} = [\mathbf{f}_{vq,1}^{e1}, \mathbf{f}_{vq,2}^{e1}, \cdots, \mathbf{f}_{vq,T}^{e1}]^\top \in \mathbb{R}^{T\times d}$. $\odot$ denotes element-wise multiplication; $[\cdot]$ is concatenation; $FFN_6(\cdot)$ is feed-forward network that projects the concatenated feature from dimension $4d$ to d.

After the cross-modal fusion network, an additional global motion learning network and an additional local motion learning network are applied to learn features for the specific actions corresponding to the language query:

$$\mathbf{F}_{e1} = GoL(LoL(\mathbf{F}_{vq}^{e1})) \tag{9}$$

where $\mathbf{F}_{e1} \in \mathbb{R}^{T\times d}$ is the output of the exploitation module; $GoL(\cdot)$ denotes the global motion learning network in Eq. (6) and $LoL(\cdot)$ denotes the local motion learning network in Eq. (7).

C. Exploration Module

Some language queries describe actions with salient postures, such as *"a person is standing holding a mirror in one hand"* and *"a person is sitting on the floor smiling"*. Features of these actions should be learned to distinguish different postures. With this in mind, we design an exploration module, which consists of a salient posture learning network and a temporal expansion network. The salient posture learning network searches several key postures of the target action, and the temporal expansion network expands the posture frames to the target action boundaries.

1) Salient Posture Learning Network: Different body postures have distinctive patterns, which can be learned by convolution layers. To handle the multiple-scale patterns, the salient posture learning network is implemented by a multi-scale temporal convolution, which has the same architecture as the local motion learning network described in Section III-B2. Specifically, given input video features $\mathbf{F}_v$, the salient posture features are learned by

$$\mathbf{F}_v^p = FFN_7([\delta(G_1(\mathbf{F}_v)), \delta(G_2(\mathbf{F}_v)), \cdots, \delta(G_K(\mathbf{F}_v))]) \tag{10}$$

where $\mathbf{F}_v^p$ represents the learned salient posture features; $FFN_7(\cdot)$ is a feed-forward network; G_j denotes temporal convolution, $j \in \{1, 2, \cdots, K\}$; $[\cdot]$ is concatenation, and the δ is $ReLU$ activation layer.

2) Temporal Expansion Network: The temporal expansion network expands the salient posture frames to action segment boundaries by learning the temporal dependency and consists of a multi-scale LSTM block and a multi-scale self-attention block. The architectures of these two blocks are the same as the global motion network described in Section III-B1. The video features $\mathbf{F}_v^{td}$ learned through the temporal expansion network are given by

$$\mathbf{F}_v^{td} = MS\text{-}LSTM(\mathbf{F}_v^p) + MSAB(\mathbf{F}_v^p) \tag{11}$$

where $MS\text{-}LSTM(\cdot)$ is same as that in Eq. (3) and $MSAB(\cdot)$ is same as that in Eq. (5), but with different parameters.

3) Cross-modal Fusion Network: Given the video features $\mathbf{F}_v^{td}$ and the language query features $\mathbf{F}_q$, We integrate them by using the same context-query attention in Section III-B3 to obtain the fused visual-linguistic features $\mathbf{F}_{vq}^{e2}$.

Then an additional salient posture learning network and an additional temporal expansion network are applied to learn features for the specific actions related to the language query:

$$\mathbf{F}_{e2} = TeN(SpL(\mathbf{F}_{vq}^{e2})) \tag{12}$$

where $\mathbf{F}_{e2} \in \mathbb{R}^{T \times d}$ is the output of the exploration module; $TeN(\cdot)$ denotes the temporal expansion network in Eq. (11); $SpL(\cdot)$ denotes the salient posture learning network in Eq. (10).

D. Query-aware Adaptive Selection Module

The exploitation and exploration modules handle various target actions with different motion characteristics and semantic granularities described by the language queries. Given a specific language query, how to dynamically and flexibly select the appropriate pathway between the two modules becomes a major problem. To address this problem, we propose a query-aware adaptive selection module for the selection between the exploitation and exploration modules according to the input language query. Specifically, the probabilities for different pathways are estimated based on the sentence-level query feature $\mathbf{F}_s$ using a fully connected layer, followed by a Gumbel-softmax normalization:

$$\mathbf{P}_{path} = Gumbel\text{-}softmax(FC_3(\mathbf{F}_s)), \tag{13}$$

where $FC_3(\cdot)$ is a fully connected layer with the output dimension 2; $\mathbf{P}_{path} \in \mathbb{R}^2$ is a two-dimension one-hot vector and denotes the probabilities of selecting the exploitation and exploration modules. Gumbel-softmax normalization is employed to make $\mathbf{P}_{path}$ a differentiable discrete sampling based on the probability. The Gumbel softmax trick is a technique that allows sampling from categorical distribution during the forward pass of a neural network. It is essentially done by combining the reparameterization trick and smooth relaxation. Given the Gumbel noise $\mathbf{g} \sim Gumbel(0, 1)$ and input $\mathbf{x}$, the soft categorical sample can be computed by Gumbel-softmax operation: $\mathbf{y} = Softmax((log(\mathbf{x}) + \mathbf{g})/\tau)$, where τ is an annealing temperature. When $\tau \to 0^+$, the output $\mathbf{y}$ is equivalent to the Gumbel-Max form: $\hat{\mathbf{y}} = Onehot(argmax(log(x) + \mathbf{x}))$. When the input $\mathbf{x}$ is unnormalized, the $log(\cdot)$ operator shall be omitted [35].

Given $\mathbf{P}_{path}$, the output feature $\mathbf{F}_e$ is given by

$$\mathbf{F}_e = \begin{cases} \mathbf{F}_{e1}, & if \quad \mathbf{P}_{path} = [1, 0]^\top \\ \mathbf{F}_{e2}, & if \quad \mathbf{P}_{path} = [0, 1]^\top \end{cases} \tag{14}$$

E. Prediction Module

We learn a probability distribution over all the video frames to represent the probabilities of the start and end boundaries of the target action segment. The distribution probabilities of the start boundary, denoted by $\mathbf{P}_s^b \in \mathbb{R}^T$, and that of the end boundary, denoted by $\mathbf{P}_e^b \in \mathbb{R}^T$, are predicted by a two-branch network consisting of two fully connected layers:

$$\begin{aligned} \mathbf{P}_s^b &= softmax(FC_5(\delta(FC_4(\mathbf{F}_e)))) \\ \mathbf{P}_e^b &= softmax(FC_7(\delta(FC_6(\mathbf{F}_e)))) \end{aligned} \tag{15}$$

where the output feature dimensions of $FC_i(\cdot), i \in \{4, 6\}$ and $FC_j(\cdot), j \in \{5, 7\}$ are $\frac{d}{2}$ and 1, respectively, and the δ is $ReLU$ activation layer.

To further improve the performance, we also apply another branch of two fully connected layers network to predict an inner probability for each frame as an auxiliary task only for training, following [34], [36]. 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 the target action frames, calculated by

$$\mathbf{P}^{in} = sigmoid(FC_9(\delta(FC_8(\mathbf{F}^{vm})))) \tag{16}$$

where the output feature dimensions of $FC_8(\cdot)$ and $FC_9(\cdot)$ are $\frac{d}{2}$ and 1, respectively, and the δ is $ReLU$ activation layer.

During testing, the predicted start and end boundaries of the target action segment are derived by maximizing the joint probability:

$$\begin{aligned} (\hat{\tau_s}, \hat{\tau_e}) &= \arg\max_{t_s, t_e} \mathbf{P}_s^b(t_s) \times \mathbf{P}_e^b(t_e) \\ p_{se}^b &= \mathbf{P}_s^b(\hat{\tau_s}) \times \mathbf{P}_e^b(\hat{\tau_e}) \end{aligned} \tag{17}$$

where p_{se}^b is the optimized probability score of the predicted boundaries $(\hat{\tau_s}, \hat{\tau_e})$.

F. Training Objective

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

$$\mathcal{L}_{bound} = f_{XE}(\mathbf{P}_s^b, \tau_s) + f_{XE}(\mathbf{P}_e^b, \tau_e) \tag{18}$$

TABLE I
COMPARISON WITH THE STATE-OF-THE-ART METHODS ON THE CHARADES-STA DATASET.

Methods	$R@1; IoU \geq \mu$			$mIoU$
	0.3	0.5	0.7	
VSLNet [3]	70.46	54.19	35.22	50.02
LGI [28]	72.96	59.46	35.48	51.38
DeNet [49]	-	59.7	38.52	-
SS [50]	-	60.75	36.19	-
I^{2}N [4]	-	56.61	34.14	-
CPNet [29]	71.94	60.27	38.74	52.00
ACRM [51]	73.47	57.53	38.33	-
ICG [52]	67.63	50.24	32.88	48.02
CPN [53]	68.48	51.07	31.54	48.08
SeqPAN [27]	73.84	60.86	41.34	53.92
CBLN [54]	-	61.13	38.22	-
MGPN [30]	-	60.82	41.16	-
EAMAT [34]	74.19	61.69	41.96	54.45
Ours	**74.43**	**62.20**	**42.52**	**55.03**

TABLE II
COMPARISON WITH THE STATE-OF-THE-ART METHODS ON THE TACOS DATASET.

Methods	$R@1; IoU \geq \mu$			$mIoU$
	0.3	0.5	0.7	
BPNet [58]	25.96	20.96	14.08	19.53
VSLNet [3]	29.61	24.27	20.03	24.11
I^{2}N [4]	31.80	28.69	-	-
SS [50]	41.33	29.56	-	-
CPNet [29]	42.61	28.29	-	28.69
CBLN [54]	38.89	27.65	-	-
ICG [52]	38.84	29.07	19.05	28.26
SMIN [36]	48.01	35.24	-	-
CPN [53]	48.29	36.58	21.25	34.63
MGPN [30]	48.81	36.74	-	-
EAMAT [34]	50.11	38.16	26.82	36.43
Ours	**53.79**	**41.19**	**28.23**	**38.59**

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 is formulated by

$$\mathcal{L}_{in} = f_{BXE}(\mathbf{P}^{in}, \mathbf{Y}^{in}) \tag{19}$$

where $f_{BXE}(\cdot)$ is a binary cross-entropy function. $\mathbf{Y}^{in} = \{y_i^{in}\}_{i=1}^{T} \in \{0,1\}$ is the ground-truth probability of each frame being target action frame, and when $\tau_s \leq i \leq \tau_e$, $y_i^{in} = 1$, otherwise $y_i^{in} = 0$. The overall objective is given by

$$\mathcal{L} = \lambda_1 \mathcal{L}_{bound} + \lambda_2 \mathcal{L}_{in} \tag{20}$$

where λ_1 and λ_2 are hyper-parameters.

G. Discussion

The proposed method strategically categorizes actions into two types and dynamically selects suitable modules for feature learning, and this strategy has the potential application to other video and language related tasks, such as Language-Guided Video Segmentation [37]–[39], Visual Reasoning [40]–[43], and Video Captioning [44]–[48]. In these tasks, extensive designed modules have achieved promising performance. However, these modules often overlook the nuanced and varied characteristics of actions, which require distinct neural architectures for precise modeling. The proposed dynamic pathway feature learning strategy provides an opportunity to enhance existing methods by integrating individual feature learning modules tailored to different action types, making it a potential generative method in video understanding.

IV. EXPERIMENTS

A. Datasets

Charades-STA. The Charades-STA dataset is built on the Charades dataset [55] and contains 6,672 daily life videos. The average duration of the videos is 29.76 seconds. There are about 2.4 annotated segments per video, whose average duration is 8.2 seconds. The whole dataset contains 16,128 samples (i.e., pairs of query and action segment), and 12,408 samples are split into the training set, and 3,720 samples are into the testing set.

TACoS. The TACoS dataset is built on the MPII Cooking Compositive dataset [56], which consists of 127 videos with an average length of 4.79 minutes. There are around 148 annotated segments per video. The whole dataset contains 18,818 samples, including 10,146 for training, 4,589 for validation, and 4,083 for testing.

B. Metrics

We adopt two metrics for the performance evaluation: 1) $R@n; IoU \geq \mu$, which denotes the recall of top-n predictions at various thresholds of the temporal Intersection over Union (IoU). It measures the percentage of predictions that have IoU with ground truth larger than the threshold μ; 2) mean averaged IoU (mIoU), which denotes the average IoU over all the test samples. We set $n = 1$ and $\mu \in \{0.3, 0.5, 0.7\}$.

C. Implementation Details

For language queries, we truncate each input sentence to have a maximum of 30 words. We use C3D [32] for the TACoS dataset and I3D [33] for the Charades-STA dataset to extract video features. Adam [57] is adopted for optimization with an initial learning rate of 5e-4, a linear decay schedule, and 50 maximum epochs. The loss weights λ_1 and λ_2 in Equation 20 are set to 1 and 10, respectively. The feature dimension d is set to 512, the head number of multi-head self-attention is set to 8, the scale number of LSTM blocks is set to 3, and the kernel sizes of temporal convolutional blocks are set to 3, 5, and 7. The numbers of LSTM Blocks and temporal convolutional blocks in both the exploitation module and the exploration module are set to 1 and 3, respectively. The kernel sizes in Eq.(7) are set to 3, 5, 7, and 9.

D. Comparison with Other Methods

We compare our method with state-of-the-art methods. These methods include both proposal-based methods (SS [50], I^{2}N [4], MAST [24], SMIN [36], CBLN [54], ICG [52], MGPN [30]) and proposal-free methods (ACRM [51], VSLNet [3], LGI [28], SeqPAN [27], CPN [53], DeNet [49], CPNet [29], EAMAT [34]).

TABLE III
ABLATION STUDIES ON THE CHARADES-STA DATASET.

Methods	$R@1; IoU \geq \mu$ 0.3	0.5	0.7	$mIoU$
exploitation module only	73.23	59.36	40.68	52.29
exploration module only	73.63	59.85	41.40	53.47
exploitation w/o LML	70.91	55.43	33.46	49.98
exploration w/o TE	69.48	54.00	36.23	50.39
network parallel	72.95	58.68	38.14	52.57
Sum fusion	71.99	58.63	38.62	51.96
Multiply fusion	73.06	58.22	39.16	53.27
Concatenate fusion	73.57	60.88	41.22	54.14
Soft dynamic	**74.65**	61.18	41.53	54.34
Ours	74.43	**62.20**	**42.52**	**55.03**

TABLE IV
ABLATION STUDIES ON THE TACOS DATASET.

Methods	$R@1; IoU \geq \mu$ 0.3	0.5	0.7	$mIoU$
exploitation module only	52.88	40.03	27.24	37.64
exploration module only	51.86	40.01	26.99	37.04
exploitation w/o LML	40.98	29.21	18.52	28.84
exploration w/o TE	44.71	35.41	24.46	33.52
network parallel	48.98	38.28	25.96	35.82
Sum fusion	51.71	39.74	27.87	37.44
Multiply fusion	51.26	39.31	26.66	37.29
Concatenate fusion	52.28	40.78	27.91	38.31
Soft dynamic	**54.04**	**42.29**	27.64	**38.98**
Ours	53.79	41.19	**28.23**	38.59

Table I and Table II show the comparison results on the Charades-STA and TACoS datasets, respectively. It is interesting to observe the promising performance improvements of our method in terms of all evaluation metrics on both two datasets, clearly validating the superiority of the query-aware dynamic pathway between different modules on video feature learning to localize actions. Specifically, our method improves the "$R@1; IoU \geq \mu$" by 0.28%, 0.51% and 0.52% on Charades-STA, respectively, when IoU threshold μ is 0.3, 0.5 and 0.7. On TACoS, the improvements of of "$R@1; IoU \geq \mu$" are 3.68%, 3.03% and 1.41%, respectively, when μ is 0.3, 0.5 and 0.7. As for $mIoU$, the gains of 0.58% and 2.16% are achieved for the Charades-STA dataset and TACoS dataset, respectively. The TACoS dataset contains various language queries and different variable-length target moments in videos, which makes it more challenging. Our method achieves more improvements on TACoS than Charades-STA, which further demonstrates the superiority of our method in complex situations.

E. Ablation Studies

We perform in-depth analysis to evaluate each component of our method on the Charades-STA and TACoS datasets, and the results are shown in Table III and Table IV, respectively.

1) Effect of the exploitation/exploration module.: The proposed dynamic pathway is a dynamic selection between an exploitation module and an exploration module to handle different kinds of actions. To evaluate the effectiveness of the dynamic pathway, we need to first verify the performance of the exploitation module and the exploration module separately, without the dynamic pathway. The results are shown in the first part of Table III and Table IV. We observe that the exploitation module achieves 59.36% and 40.03% in terms of "$R@1; IoU \geq 0.5$" on the Charades-STA dataset and TACoS dataset, respectively, and the exploration module achieves 59.85% and 40.01% in terms of "$R@1; IoU \geq 0.5$" on the Charades-STA dataset and TACoS dataset, respectively. Both the exploitation module and the exploration module perform comparably or better than some of the existing methods (*e.g.*,VSLNet [3],CPN [53]), which demonstrates the effectiveness of the exploitation module on handling the actions with complex motion relationships and the exploration module on handing the actions with salient postures.

2) Effect of the local motion learning network: To evaluate the effect of the local motion learning network, we replace the local motion learning network in Section III-B2 with the global motion learning network in Section III-B1 for comparison, denoted as "exploitation w/o LML". From Table I and Table II, From Table III and Table IV, it is obvious that the local motion learning network, the performance of all evaluation metrics drops significantly on both datasets which demonstrates the benefit of learning subtle motion changes to improve the accuracy of boundary prediction.

3) Effect of the temporal expansion network: To evaluate the effect of the temporal expansion network, we replace the temporal expansion network in Section III-C2 with the salient posture learning network in Section III-C1, denoted as "exploration w/o TE". From Table III and Table IV, we also observe a large drop in performance without the temporal expansion network, which suggests it is important to learn the temporal dependency to expand posture frames to the action segment.

4) Effect of different orders of networks: Both the exploitation and exploration modules consist of sequential networks. Specifically, the exploitation module has a sequence of a global motion learning network and a local motion learning network. The exploration module has a sequence of a salient posture learning network and a temporal expansion network. To analyze the effect of different network orders, we stack the networks in parallel, denoted as "network parallel", as shown in the second part of Table III and Table IV. It is interesting to observe that compared with the exploitation module and exploration module, "network parallel" achieves worse results, which demonstrates the superiority of the coarse-to-fine manner in the exploitation module and the point-to-area manner in the exploration module on learning video features.

5) Effect of the dynamic pathway.: As shown in the last row "Ours" in Table III and Table IV, compared with the "exploitation module" and "exploration module", the performance is improved by adaptively selecting the exploitation module or the exploration module based on the language query for each input, which verifies the effectiveness of the proposed query-aware adaptive selection module.

To further evaluate the effectiveness of the dynamic pathway, we also design several methods of fusing the features learned by the exploitation module and the exploration module for comparison, denoted as "Sum fusion", "Multiply fusion" and "Concatenate fusion". The "Sum fusion" indicates that features from the exploitation module and the exploration

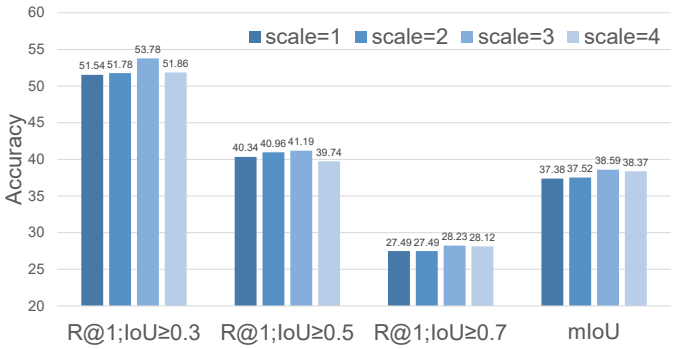


Fig. 4. Results of different scale numbers in LSTM on the TACoS dataset.

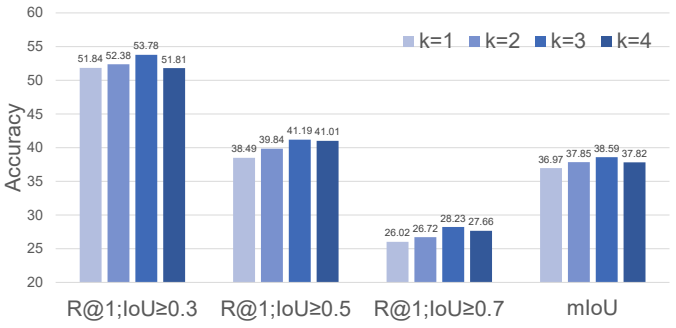


Fig. 5. Results of different scale numbers in temporal convolution on the TACoS dataset.

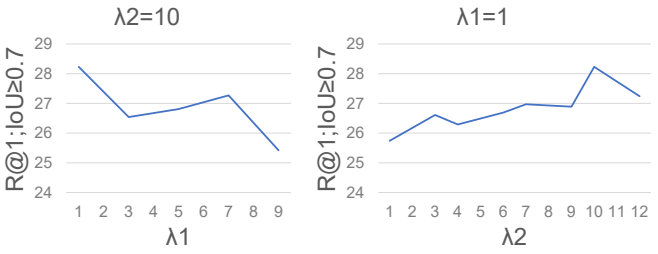


Fig. 6. Results of different loss weights on the TACoS dataset.

module are summarized, and "Multiply fusion" indicates that features from the exploitation module and the exploration module are element-wise multiplied. The "Concatenate fusion" means we concatenate the features of the exploitation module and the exploration module, and "Soft dynamic" means we sum the features of the exploitation module and the exploration module based on their probabilities, which is implemented by replacing Gumble-softmax with softmax of Eq.(13).

As shown in the third part of Table III and Table IV, we observe that these fusion methods achieve little improvement or even worse performance, compared with the exploitation module or the exploration module, probably due to the redundant information introduced by the exploitation module and exploration module in feature learning, thus leading to inferior performance. In contrast, our dynamic pathway succeeds in selecting the appropriate pathway between the exploitation module and exploration module according to the specific language query, thus contributing to further improvements.

From Table IV, we observe that "Soft dynamic" achieves better results on most evaluation metrics on the TACoS dataset, probably due to the intricate and diverse action samples. However, the improvements are marginal compared with "Ours". Conversely, on the Charades-STA dataset in Table III, the "Soft dynamic" achieves worse performance than "Ours", and the possible reason is that the redundant module brings noise. Furthermore, the "Soft dynamic" necessitates the simultaneous execution of both pathways during inference, which results in double memory consumption and increased runtime, making it less practical in real-world applications.

Discussion: We classify all actions into two classes: action with complex motion relationships and action with salient body posture, and adopt the exploitation module and exploration module to learn features. For a certain sample, the two modules are adaptively selected based on their input language query. When the query actions are none of the two classes, both the exploitation module and exploration module get sub-optimal results, but the better one may be chosen. Likewise, when the query action have both the complex motion and salient body posture, both the exploitation module and exploration module get fine results and the better one may be chosen. Our dynamic pathway strategy navigates within the constrained solution space to identify the better pathway.

F. Parameter Analysis

1) Temporal Scale Number of LSTM: To analyze the effect of the temporal scale number of LSTM (i.e., S in Eq. (3)) on the performance, we conduct experiments by using different numbers of temporal scale, and the results are shown in Figure 4. As the scale number increases, the performance first increases gradually and then decreases, which suggests that multiple temporal scales of LSTM can handle various duration of the same actions, but too many scales may introduce redundant information not related to the target action.

2) Temporal Scale Number of Convolution: To evaluate the effect of temporal scale number of convolution (i.e., K in Eq. (7)), different numbers of kernels are applied, and the results are shown in Figure 5. It is obvious that more different kernels help improve the accuracy, but too many kernels will introduce noisy information to hurt the performance.

3) Loss Weight: To further evaluate the effectiveness of the training losses, we tune the hyper-parameters λ_1 and λ_2 in Eq. (20) in the range of $[1, 12]$. Figure 6 shows the results of different values of λ_1 and λ_2. We observe that the performance drops as λ_1 increases. In contrast, the performance improves along with the increasing λ_2 until reaches the maximum. It is interesting that the value of λ_2 is 10 times that of λ_1, which suggests that the per-frame of inner action prediction gives strong support for boundary prediction.

G. Qualitative Results

We visualize several examples of action localization results in Figure 7, where "Ours (exploitation module)" denotes the results predicted by the exploitation module, "Ours (exploration module)" denotes the results predicted by the exploration module, and "✓" denotes the selection path, estimated by the query-aware adaptive selection module.

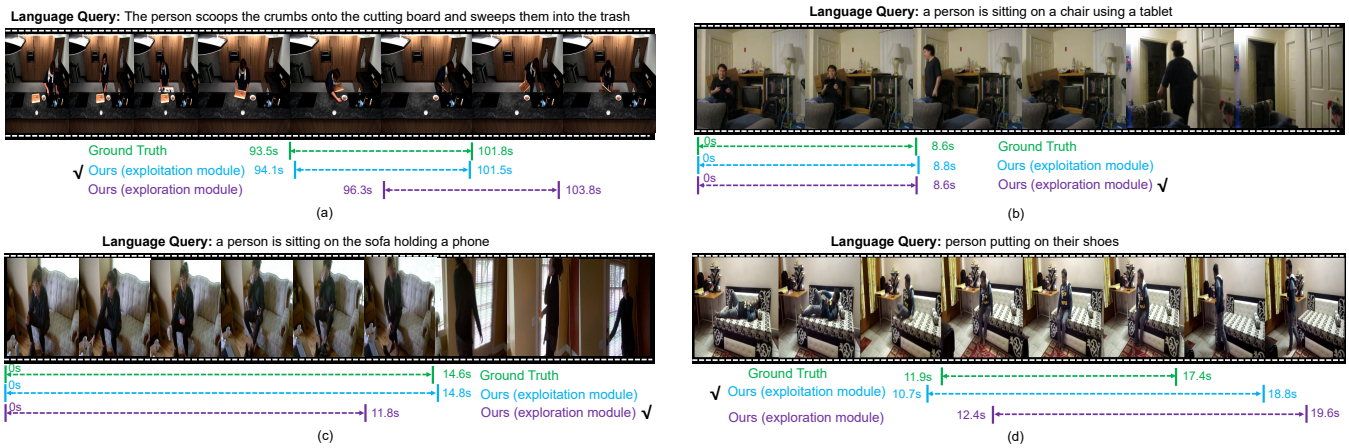


Fig. 7. Examples of action localization results. "Ours (exploitation module)" denotes the results predicted by the exploitation module, "Ours (exploration module)" denotes the results predicted by the exploration module, and "✓" denotes the selection path, estimated by the query-aware adaptive selection module.

For the first example shown in Figure 7 (a), the action "*The person scoops the crumbs onto the cutting board and sweeps them into the trash*" contains complex motion relationships, and our method successfully searches the action segment by selecting the appropriate exploitation module.

For the second example shown in Figure 7 (b), the action "*a person is sitting on a chair using a tablet*" has several salient body postures that are distinguishable for classification, so the exploration module is selected for localization in our method. Furthermore, it is interesting to observe that both the exploitation module and the exploration module achieve good results, since this action is simple and easy to be localized.

For the third example shown in Figure 7 (c), we observe that the exploitation module predicts the action boundaries more closer to the ground truth. But our method selects the exploration module, which estimates a wrong ending boundary of the target action segment compared with the ground truth. To go further, we find that the ending boundary of the ground truth segment is wrongly located at the time when the person is standing. So the ground truth annotation is biased, however, the selected exploration module still succeeds in capturing the action of "*sitting on sofa holding a phone*" and estimating a right truly ending boundary of the action, which further demonstrates the necessity that different actions should be handled by different modules.

For the fourth example shown in Figure 7 (d), both the exploitation module and the exploration module predict wrong boundaries of the target action "*putting shoes*", making it useless of the dynamic pathway selection of our method. This demonstrates the weakness of our methods on fine-grained feature learning that requires finding small clues on the complex backgrounds. For the fourth example shown in Figure 7 (d), both the exploitation module and the exploration module do not accurately predict the boundaries of the target action "*putting shoes*", so the dynamic selection fails. This demonstrates the weakness of our methods on the fine-grained action localization, which requires capturing fine-grained visual and motion clues for prediction.

V. Conclusion

We have presented a novel dynamic pathway method between different modules to learn query-aware video features for language-driven action localization. Our method succeeds in flexibly and adaptively selecting a better feature learning path between an exploitation module and an exploration module for different kinds of actions described by the language queries. The exploitation module works in a coarse-to-refine manner and is able to handle the actions with coarse semantic granularities and complex motion changes of multiple sub-actions. The exploration module functions in a point-to-region diffusion fashion and is able to deal with the actions with fine semantic granularities and salient body postures. Experiments on the Charades-STA dataset and TACoS dataset validate the effectiveness of our method. In future work, we plan to design a more flexible dynamic architecture using activate learning to actively search appropriate modules.

Acknowledgments

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

References

[1] Y. Yuan, T. Mei, and W. Zhu, "To find where you talk: Temporal sentence localization in video with attention based location regression," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 33, no. 1, 2019, pp. 9159–9166.

[2] D. Liu, X. Qu, X.-Y. Liu, J. Dong, P. Zhou, and Z. Xu, "Jointly cross- and self-modal graph attention network for query-based moment localization," in *Proceedings of the 28th ACM International Conference on Multimedia (ACM MM)*, 2020, pp. 4070–4078.

[3] H. Zhang, A. Sun, W. Jing, L. Zhen, J. T. Zhou, and R. S. M. Goh, "Natural language video localization: A revisit in span-based question answering framework," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 8, pp. 4252–4266, 2021.

[4] K. Ning, L. Xie, J. Liu, F. Wu, and Q. Tian, "Interaction-integrated network for natural language moment localization," *IEEE Transactions on Image Processing*, vol. 30, pp. 2538–2548, 2021.

[5] J. Gao, C. Sun, Z. Yang, and R. Nevatia, "Tall: Temporal activity localization via language query," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2017, pp. 5277–5285.

[6] L. A. Hendricks, O. Wang, E. Shechtman, J. Sivic, T. Darrell, and B. Russell, "Localizing moments in video with natural language," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2017, pp. 5804–5813.

[7] J. Chen, L. Ma, X. Chen, Z. Jie, and J. Luo, "Localizing natural language in videos," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 33, no. 01, 2019, pp. 8175–8182.

[8] D. Cao, Y. Zeng, X. Wei, L. Nie, R. Hong, and Z. Qin, "Adversarial video moment retrieval by jointly modeling ranking and localization," in *Proceedings of the 28th ACM International Conference on Multimedia (ACM MM)*, 2020, pp. 898–906.

[9] J. Chen, X. Chen, L. Ma, Z. Jie, and T.-S. Chua, "Temporally grounding natural sentence in video," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2018, pp. 162–171.

[10] Y. Yuan, L. Ma, J. Wang, W. Liu, and W. Zhu, "Semantic conditioned dynamic modulation for temporal sentence grounding in videos." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2020.

[11] S. Zhang, H. Peng, J. Fu, and J. Luo, "Learning 2d temporal adjacent networks for moment localization with natural language." in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 34, no. 7, 2020, pp. 12 870–12 877.

[12] S. Chen and Y.-G. Jiang, "Semantic proposal for activity localization in videos via sentence query," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 33, no. 1, 2019, pp. 8199–8206.

[13] J. Wang, L. Ma, and W. Jiang, "Temporally grounding language queries in videos by contextual boundary-aware prediction," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 34, no. 7, 2020, pp. 12 168–12 175.

[14] A. Wu and Y. Han, "Multi-modal circulant fusion for video-to-language and backward." in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI)*, 2018, pp. 1029–1035.

[15] H. Wang, Z.-J. Zha, X. Chen, Z. Xiong, and J. Luo, "Dual path interaction network for video moment localization," in *Proceedings of the 28th ACM International Conference on Multimedia (ACM MM)*, 2020, pp. 4116–4124.

[16] X. Qu, P. Tang, Z. Zou, Y. Cheng, J. Dong, P. Zhou, and Z. Xu, "Fine-grained iterative attention network for temporal language localization in videos," in *Proceedings of the 28th ACM International Conference on Multimedia (ACM MM)*, 2020, pp. 4280–4288.

[17] S. Yang, Z. Shang, and X. Wu, "Probability distribution based frame-supervised language-driven action localization," in *Proceedings of the 31st ACM International Conference on Multimedia*, 2023, pp. 5164–5173.

[18] R. Zeng, H. Xu, W. Huang, P. Chen, M. Tan, and C. Gan, "Dense regression network for video grounding," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 10 287–10 296.

[19] C. Rodriguez-Opazo, E. Marrese-Taylor, F. S. Saleh, H. Li, and S. Gould, "Proposal-free temporal moment localization of a natural-language query in video using guided attention," in *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2020, pp. 2464–2473.

[20] M. Hahn, A. Kadav, J. M. Rehg, and H. P. Graf, "Tripping through time: Efficient localization of activities in videos," in *BMVC*, 2019.

[21] C. Lu, L. Chen, C. Tan, X. Li, and J. Xiao, "Debug: A dense bottom-up grounding approach for natural language video localization," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 5143–5152.

[22] J. Wu, G. Li, S. Liu, and L. Lin, "Tree-structured policy based progressive reinforcement learning for temporally language grounding in video," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 34, no. 7, 2020, pp. 12 386–12 393.

[23] D. Liu, X. Fang, P. Zhou, X. Di, W. Lu, and Y. Cheng, "Hypotheses tree building for one-shot temporal sentence localization," *ArXiv*, vol. abs/2301.01871, 2023.

[24] M. Zhang, Y. Yang, X. Chen, Y. Ji, X. Xu, J. Li, and H. T. Shen, "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)*, 2021, pp. 12 669–12 678.

[25] D. Liu and W. Hu, "Skimming, locating, then perusing: A human-like framework for natural language video localization," in *Proceedings of the 30th ACM International Conference on Multimedia (ACM MM)*, 2022, pp. 4536–4545.

[26] S. Ghosh, A. Agarwal, Z. Parekh, and A. G. Hauptmann, "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: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 1984–1990.

[27] H. Zhang, A. Sun, W. Jing, L. Zhen, J. T. Zhou, and S. M. R. Goh, "Parallel attention network with sequence matching for video grounding," in *ACL 2021: 59th annual meeting of the Association for Computational Linguistics (ACL)*, 2021, pp. 776–790.

[28] J. Mun, M. Cho, and B. Han, "Local-global video-text interactions for temporal grounding," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 10 810–10 819.

[29] K. Li, D. Guo, and M. Wang, "Proposal-free video grounding with contextual pyramid network." in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 35, no. 3, 2021, pp. 1902–1910.

[30] X. Sun, X. Wang, J. Gao, Q. Liu, and X. Zhou, "You need to read again: Multi-granularity perception network for moment retrieval in videos," *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (ACM SIGIR)*, 2022.

[31] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1532–1543.

[32] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE/CVF international conference on computer vision (ICCV)*, 2015, pp. 4489–4497.

[33] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6299–6308.

[34] S. Yang and X. Wu, "Entity-aware and motion-aware transformers for language-driven action localization," in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI)*, L. D. Raedt, Ed. ijcai.org, 2022, pp. 1552–1558.

[35] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," in *International Conference on Learning Representations (ICLR)*, 2017.

[36] H. Wang, Z.-J. Zha, L. Li, D. Liu, and J. Luo, "Structured multi-level interaction network for video moment localization via language query," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 7026–7035.

[37] K. Gavrilyuk, A. Ghodrati, Z. Li, and C. G. Snoek, "Actor and action video segmentation from a sentence," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 5958–5966.

[38] C. Liang, W. Wang, T. Zhou, J. Miao, Y. Luo, and Y. Yang, "Local-global context aware transformer for language-guided video segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[39] T. Hui, S. Liu, Z. Ding, S. Huang, G. Li, W. Wang, L. Liu, and J. Han, "Language-aware spatial-temporal collaboration for referring video segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[40] J. Hou, X. Wu, X. Zhang, Y. Qi, Y. Jia, and J. Luo, "Joint commonsense and relation reasoning for image and video captioning," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 07, 2020, pp. 10 973–10 980.

[41] C. Liang, W. Wang, T. Zhou, and Y. Yang, "Visual abductive reasoning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 15 565–15 575.

[42] J. Chen, X. Wu, Y. Hu, and J. Luo, "Spatial-temporal causal inference for partial image-to-video adaptation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 2, 2021, pp. 1027–1035.

[43] J. Li, P. Wei, W. Han, and L. Fan, "Intentqa: Context-aware video intent reasoning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 11 963–11 974.

[44] A. Wu, Y. Han, Y. Yang, Q. Hu, and F. Wu, "Convolutional reconstruction-to-sequence for video captioning," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 11, pp. 4299–4308, 2019.

[45] A. Wu, Y. Han, and Y. Yang, "Video interactive captioning with human prompts." in *IJCAI*, 2019, pp. 961–967.

[46] A. Wu, Y. Han, Z. Zhao, and Y. Yang, "Hierarchical memory decoder for visual narrating," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 6, pp. 2438–2449, 2020.

[47] W. Zhao, X. Wu, and J. Luo, "Multi-modal dependency tree for video captioning," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 34, pp. 6634–6645, 2021.
[48] A. Yang, A. Nagrani, P. H. Seo, A. Miech, J. Pont-Tuset, I. Laptev, J. Sivic, and C. Schmid, "Vid2seq: Large-scale pretraining of a visual language model for dense video captioning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 10 714–10 726.
[49] H. Zhou, C. Zhang, Y. Luo, Y. Chen, and C. Hu, "Embracing uncertainty: Decoupling and de-bias for robust temporal grounding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 8445–8454.
[50] X. Ding, N. Wang, S. Zhang, D. Cheng, X. Li, Z. Huang, M. Tang, and X. Gao, "Support-set based cross-supervision for video grounding," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 11 573–11 582.
[51] H. Tang, J. Zhu, M. Liu, Z. Gao, and Z. Cheng, "Frame-wise cross-modal matching for video moment retrieval," *IEEE Transactions on Multimedia*, vol. 24, pp. 1338–1349, 2020.
[52] G. Nan, R. Qiao, Y. Xiao, J. Liu, S. Leng, H. Zhang, and W. Lu, "Interventional video grounding with dual contrastive learning." in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 2765–2775.
[53] Y. Zhao, Z. Zhao, Z. Zhang, and Z. Lin, "Cascaded prediction network via segment tree for temporal video grounding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 4197–4206.
[54] D. Liu, X. Qu, J. Dong, P. Zhou, Y. Cheng, W. Wei, Z. Xu, and Y. Xie, "Context-aware biaffine localizing network for temporal sentence grounding." in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 11 235–11 244.
[55] G. A. Sigurdsson, G. Varol, X. Wang, A. Farhadi, I. Laptev, and A. Gupta, "Hollywood in homes: Crowdsourcing data collection for activity understanding," in *European Conference on Computer Vision (ECCV)*. Springer, 2016, pp. 510–526.
[56] M. Rohrbach, M. Regneri, M. Andriluka, S. Amin, M. Pinkal, and B. Schiele, "Script data for attribute-based recognition of composite activities," in *European Conference on Computer Vision (ECCV)*. Springer, 2012, pp. 144–157.
[57] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
[58] S. Xiao, L. Chen, S. Zhang, W. Ji, J. Shao, L. Ye, and J. Xiao, "Boundary proposal network for two-stage natural language video localization," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*. AAAI Press, 2021, pp. 2986–2994.