Multi-stage Aggregated Transformer Network for Temporal Language Localization in Videos

Mingxing Zhang[1], Yang Yang[1,2,*], Xinghan Chen[1], Yanli Ji[1], Xing Xu[1,2], Jingjing Li[1], Heng Tao Shen[1]
[1] School of Computer Science and Engineering and
Digital Media Technology Key Laboratory of Sichuan Province, UESTC
[2] Institute of Electronic and Information Engineering of UESTC in Guangdong
{minsingcheong, dlyyang, xinghanchen111}@gmail.com, {yanliji, xing.xu}@uestc.edu.cn,
lijin117@yeah.net, shenhengtao@hotmail.com

Abstract

We address the problem of localizing a specific moment from an untrimmed video by a language sentence query. Generally, previous methods mainly exist two problems that are not fully solved: 1) How to effectively model the fine-grained visual-language alignment between video and language query? 2) How to accurately localize the moment in the original video length? In this paper, we streamline the temporal language localization as a novel multi-stage aggregated transformer network. Specifically, we first introduce a new visual-language transformer backbone, which enables iterations and alignments among all elements in visual and language sequences. Different from previous multi-modal transformers, our backbone keeps both structure unified and modality specific. Moreover, we also propose a multi-stage aggregation module topped on the transformer backbone. In this module, we compute three stage-specific representations corresponding to different moment stages respectively, i.e. starting, middle and ending stages, for each video element. Then for a moment candidate, we concatenate the starting/middle/ending representations of its starting/middle/ending elements respectively to form the final moment representation. Because the obtained moment representation captures the stage specific information, it is very discriminative for accurate localization. Extensive experiments on ActivityNet Captions and TACoS datasets demonstrate our proposed method achieves significant improvements compared with all other methods.

1. Introduction

Temporal localization is a prominent and fundamental problem for video analysis in the computer vision community. In the past years, there are lots of works that have been conducted for temporal action localization [28, 38, 48, 3, 42, 18, 47]. Recently, the task of temporally localizing natural language in videos has been attracting the interest of researchers. The task aims to localize the temporal moment corresponding to a language sentence query in an untrimmed video. Compared with temporal action localization, temporal language localization is more challenging and has vast potential applications, such as video retrieval, video captioning, and human-computer interaction, etc.

There are many approaches that have been proposed for temporal language localization [1, 7, 20, 41, 40, 45, 35, 19, 25, 26, 43]. Although those approaches have achieved many promising results, there are still several critical problems that have not been fully solved: 1) How to effectively model the fine-grained visual-language alignment between video and language query? 2) How to accurately localize the moment in the original video length? For the first problem, most existing approaches often process video and language sequences separately and then fuse them. However, processing the two modalities separately, e.g., first encoding the query sentence into a single vector, will inevitably lose some detailed semantics and thus cannot achieve detailed interaction and alignment. Besides, the temporal relations in the video sequence are often modeled by local operations, which is not sufficient to obtain enough contextual information. For the second problem, previous approaches usually use the full convolution, mean pooling or RoI (Region of Interest) pooling [38, 10] operations to obtain the feature representation for moment candidates. We argue that these kinds of representations are not discriminative enough. For instance, the moment or event often contains some different stages, e.g. the starting, middle and ending stages. The information of those stages is very important for accurate moment localization. However, the mean pooling operation discards the stage information, thus cannot match the

[*] Corresponding author.

different stages precisely. Although the convolution or RoI pooling operations can model the different stages to some extent, they do not rely on explicit stage-specific representations. Besides, convolution operation densely using all the elements in the moment candidate is hard to catch the key elements for localization, and it also cannot adapt to various dynamics in the moment because of the fixed structure of convolution kernel.

To address these problems, in this paper we propose a novel multi-stage aggregated transformer network for temporal language localization in videos. Our proposed network mainly contains two components: the visual-language transformer backbone and the multi-stage aggregation module topped on the transformer backbone. Specifically, we first introduce a new visual-language transformer backbone, which simultaneously processes both the video and language sequences to effectively model the fine-grained visual-language interactions and alignments. Our transformer backbone is inspired by recently proposed visual-language BERT models [31, 30, 17, 16, 5], which encode the visual and language sequences into a unified sequence and process it utilizing a single BERT. This kind of architecture processes the two sequences from different modalities in a compact and efficient way. However, we argue that different modalities have modality specific contents and relation patterns. It is not optimal to encode sequences from different modalities into a unified sequence and model them without a difference. In our transformer backbone, we also keep a single BERT architecture but decouple the BERT parameters into different groups to process the visual and language contents respectively. Thus, our transformer backbone keeps the compactness and efficiency of the single BERT structure while models the two modalities more effectively. Moreover, in order to achieve more accurate moment localization, we propose a multi-stage aggregation module topped on the transformer backbone. In this module, we compute three stage-specific representations corresponding to three different temporal stages respectively, i.e. starting, middle and ending stages, for each element in the video sequence. Then for a moment candidate, we concatenate the starting representation of its starting element, middle representation of its middle element and ending representation of its ending element to form the final moment representation. Because the three representations capture the specific information about different stages respectively, the obtained moment representation is stage sensitive, which is very discriminative for accurate localization. The whole architecture of our proposed network is conceptually simple and efficient. Not only does it achieve superior localization performance, but also achieves a very fast speed.

To summarize, our main contributions are three-fold:

- We propose a novel streamline network based on a new visual-language transformer backbone for temporal language localization. In our transformer backbone, we keep a single BERT architecture but decouple the parameters into different groups to process the modality specific contents respectively. It is the first attempt of utilizing the unified cross-modal transformer network to solve the fine-grained visual language alignment problem for temporal language localization.

- We propose a multi-stage aggregation module topped on the transformer backbone for more accurate language localization. The obtained representation consists of several sparsely selected and stage specific representations, which is very discriminative for accurate moment localization.

- We conduct extensive experiments on ActivityNet Captions and TACoS datasets and the experimental results demonstrate the effectiveness of our proposed network. We believe our work will promote the future research of this new kind of architecture for temporal language localization.

2. Related Work

The research topics about the relation of vision and language have been long explored. Temporal moment localization in untrimmed videos with natural language is an important problem among these topics. In the past years, there are lots of works have been conducted for temporal action localization. Temporal action localization aims to predict the duration and the label of the activity instance in untrimmed videos. This task is limited to simple actions and cannot handle complex activities in the real world. To tackle this problem, moment localization with natural language [7, 1] is introduced recently.

Localizing moments in videos by specified sentences is a challenging task. It requires to align the semantics between video clips and sentences in addition to video content understanding. Recently, great progress has been achieved in moment localization with natural language. Existing methods for the task can be generally divided into two categories: one-stage and two-stage. One-stage methods take a video and query sentences as input to generate the video clips associated with the query sentences directly. In [41], the authors propose an approach which can directly predict the coordinates of the queried video clip using attention mechanism. [9] and [4] leverage cross-modal interactions between video and sentence to select the starting and ending frames of the video clip described by the query sentence. [43] densely predicts the boundary regression from each frame to the ground truth moment. One-stage methods can be trained end-to-end, but they still have some limitations. They are easy to miss some candidate. Meanwhile, the interaction of video and sentences is also lim-

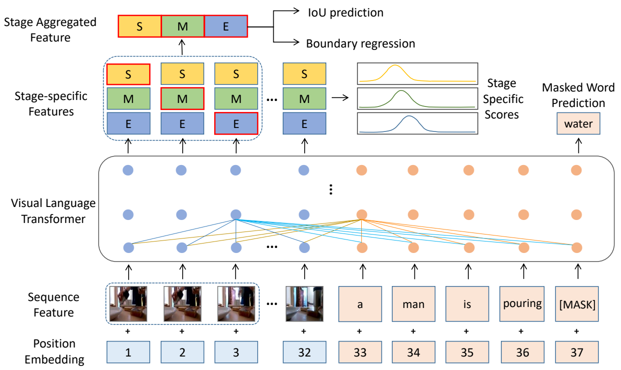


Figure 1. The framework of our proposed multi-stage aggregated transformer network for temporal language localization in videos. The tokens "[MASK]" represent the masked words. "S", "M", "E" are the representations for starting, middle and ending stages respectively. The dotted rounded rectangle represents a moment candidate.

ited. Most two-stage methods belong to a propose-and-rank pipeline. They usually generate a series of proposals from the video first, then rank these proposals relying on the matching between proposals and query sentence. Many works [7, 1, 20, 21, 46, 8, 36, 39, 44, 45] follow this pipeline. [12] introduce a reinforcement learning method into temporal language localization task, where the agent can learn the policy to adjust the boundary of a moment candidates. [39] propose a neural network that can use query to generate query-guided proposals. [44] use graph to model temporal relations among proposals explicitly. [45] propose a 2D temporal adjacent network (2D-TAN) to model the context and structure information between the moment candidates and also learn the differences between those candidates. Recently, LGI [25], CSMGAN [19] and FIAN [26] attempt to explore the local and more detailed interactions between video and sentence. However, they are well designed and not general. Besides, at the moment level, they neglect the stage specific information thus cannot achieve more accurate localization.

3. Our Method

The whole architecture of our proposed network is shown in Figure 1. Our network includes two main components: the visual-language transformer backbone and the multi-stage aggregation module topped on the backbone. Those two components are deeply integrated to form an efficient and effective network.

3.1. Problem Formulation

Given an untrimmed video, we denote the video as a sequence of frames $X = \{x_1, x_2, \cdots, x_T\}$. Each video is annotated with a set of moment-sentence pairs $\{S, t_s, t_e\}$, where S represents the language sentence, t_s and t_e represent the start and end times of the moment corresponding to the sentence. Given the input untrimmed video and the sentence query, our task aims to localize the target moment corresponding to the sentence query in the video.

To obtain the input visual feature, we first evenly segment the original video stream into N video clips. For each clip, we extract the visual features using an existing CNN model, then we mean pool the features in the clip. Thus, the video can be finally represented as $V = \{v_i\}_{i=1}^{N}$, where v_i is the feature of the i-th clip. For the input language sentence, we directly generate the Glove embedding vector w_i for each word, and thus the language sequence can be represented as $S = \{w_i\}_{i=1}^{M}$, where M is the length of the language sentence.

3.2. The Visual-language Transformer Backbone

In this section, we describe our visual-language transformer backbone in detail. Basically, our visual-language transformer is inspired by recently proposed multi-modal BERT models [31, 30, 17, 16, 5]. These models encode the visual and language sequences into a unified sequence. Then, the unified sequence is feed-forwarded to a single BERT to model the visual and language interactions. This kind of BERT is architecture concise. It can process the two modality sequences very compactly and efficiently. As shown in Figure 1, our visual-language transformer backbone also applies the single BERT architecture. However, as we argued before, different modalities have modality specific contents and relation patterns. It is not optimal to encode different modalities into a unified sequence and model them without any difference. Thus, different from those models, we decouple the parameters in the BERT into different groups to process the visual and language contents respectively. In our transformer backbone, we keep both structure unified and modality specific. Specifically, we first project the input visual and sentence features to the same dimension, then directly add the position embeddings to the feature sequences to form the input of our transformer backbone. Note that the visual and sentence sequences are concatenated as a single long sentence when encoding the element position. Let $x^l = \{v_1^l, v_2^l, \cdots, v_N^l, w_1^l, w_2^l, \cdots, w_M^l\}$ be the features of l-th layer (x^0 is the input of the backbone), the forward process in $l+1$-th layer is illustrated as follows:

$$\begin{cases} \tilde{a}_{i,V\to V}^{l+1} = Att\left(Q_{V\to V}^{l+1} v_i^l, K_{V\to V}^{l+1}\left[v_1^l, v_2^l, \cdots, v_N^l\right]\right), \\ \tilde{a}_{i,L\to V}^{l+1} = Att\left(Q_{L\to V}^{l+1} v_i^l, K_{L\to V}^{l+1}\left[w_1^l, w_2^l, \cdots, w_M^l\right]\right), \\ a_{i,V}^{l+1} = Softmax\left(\left[\tilde{a}_{i,V\to V}^{l+1}, \tilde{a}_{i,L\to V}^{l+1}\right]\right), \\ \tilde{v}_{i,V\to V}^{l+1} = W_{V\to V}^{l+1}\left[v_1^l, v_2^l, \cdots, v_N^l\right], \\ \tilde{v}_{i,L\to V}^{l+1} = W_{L\to V}^{l+1}\left[w_1^l, w_2^l, \cdots, w_M^l\right], \\ v_i^{l+1} = \left[\tilde{v}_{i,V\to V}^{l+1}, \tilde{v}_{i,L\to V}^{l+1}\right]\left(a_{i,V}^{l+1}\right)^T \end{cases} \quad (1)$$

where V, L represent the visual and language modalities respectively, Q, K, W are the learnable parameters (in which different subscripts represent different parameters), $[\cdot]$ is the concatenation operation, $Att\left(\cdot\right)$ is the attention function which is same as the original transformer model [34]. In equation 1, we decouple different modalities by using different parameters. w_i^{l+1} can be computed by the similar process. Then, we can obtain the features of $l+1$-th layer $x^{l+1}=\left\{v_1^{l+1}, v_2^{l+1}, \cdots, v_N^{l+1}, w_1^{l+1}, w_2^{l+1}, \cdots, w_M^{l+1}\right\}$. By decoupling different modalities, modality specific contents and relation patterns can be better modeled in our transformer backbone.

From equation 1, we can also see that we do not introduce any extra computation compared with previous single BERT based models [31, 30, 17, 16, 5]. The architecture is not changed and we just use different parameters to process different modality contents. Thus, we both keep the compactness and efficiency of these models and enhance the multi-modality modeling ability. Note that our transformer backbone is also different from other multi-modal BERT models [24, 32] which use two BERT streams to process different modality contents. These two BERT based models introduce additional cross-modal layers to achieve multi-modality interactions, while our transformer backbone keeps the same architecture of original BERT model which is more compact and efficient.

Our transformer backbone consists of multiple layers of this encoder. After stacking multiple layers, the derived representation is of rich capability in aggregating and aligning visual-language clues. Each element in the video can interact with each element in the sentence query. Thus, it can achieve a more detailed and accurate video-query alignment, which is very important for accurate moment localization.

3.3. Multi-stage Aggregation Module

After the visual-language transformer backbone, the derived representation is much informative and representative. However, in order to achieve more accurate moment localization performance, we propose a multi-stage aggregation module topped on the transformer backbone. In this module, we compute three stage-specific representations corresponding to different temporal stages respectively, i.e. starting, middle and ending stages, for each element in the video sequence. To improve the discrimination of representations for different stages, we also impose a prediction layer on those representations to predict the starting, middle and ending scores respectively. Those processes can be illustrated as follows:

$$\begin{cases} r^o = MLP_1^o\left(v^R\right) \\ p^o = Sigmoid\left(MLP_2^o\left(r^o\right)\right) \end{cases}, \quad o \in \{s, m, e\} \tag{2}$$

where v^R is the output visual representation from the transformer backbone, r^s, r^m, r^e are the representations for starting, middle and ending stages respectively, p^s, p^m, p^e are the prediction scores if the element is the starting, middle and ending stages respectively. For each moment-sentence pair $\{S, t_s, t_e\}$, we define the ground truth of starting, middle and ending scores as follows:

$$\begin{cases} g^s = e^{-\frac{(i-t_s)^2}{2\sigma_s^2}}, \\ g^m = e^{-\frac{(i-(t_s+t_e)/2)^2}{2\sigma_m^2}}, \\ g^e = e^{-\frac{(i-t_e)^2}{2\sigma_s^2}} \end{cases} \tag{3}$$

where i is the position index in the video sequence, g^s, g^m, g^e are the ground truth of starting, middle and ending scores, $\sigma_o = \alpha_o\left(t_e - t_s\right),\ o \in \{s, m\}$ is the standard deviation of the unnormalized 2D Gaussian distribution and α_o is a positive scalar to control the value of the standard deviation. A larger α_o makes elements near around the starting/middle/ending point of the ground truth moment have higher starting/middle/ending scores.

Then we impose a weighted binary cross entropy loss on the prediction layer as:

$$L_{stage} = \frac{1}{3}\sum_{o}^{s,m,e}\sum_{i=1}^{N}\left(-g_i^o \cdot log\left(p_i^o\right) - \left(1-g_i^o\right) \cdot log\left(1-p_i^o\right)\right) \cdot \left(g_i^o - p_i^o\right)^2 \tag{4}$$

By this loss, the three representations, i.e. r^s, r^m, r^e, will be forced to be stage specific.

Given a moment candidate with the temporal location $({t_s}', {t_e}')$, we sparsely sample its starting element (localizing at $\lfloor {t_s}' \rfloor$), middle element (localizing at $\lfloor ({t_s}' + {t_e}')/2 \rfloor$) and ending element (localizing at $\lceil {t_e}' \rceil$), and take the starting, middle and ending representations from those three elements respectively to concatenate them as the representation for the moment candidate. Then, we use this concatenated feature to predict the matching score and boundary regression between the moment candidate and ground truth moment, which is as follows:

$$a', {r_s}', {r_e}' = MLP\left(\left[r^s_{\lfloor {t_s}' \rfloor}, r^m_{\lfloor ({t_s}'+{t_e}')/2 \rfloor}, r^e_{\lceil {t_e}' \rceil}\right]\right) \tag{5}$$

where $a', {r_s}', {r_e}'$ are the predicted matching score and boundary regression, $\lfloor \cdot \rfloor$ represents the rounding down function and $\lceil \cdot \rceil$ represents the rounding up function, $[\cdot]$ represents the concatenation operation. Because the three representations are stage-specific to the moment starting, middle and ending stages, this concatenated feature is very discriminative for accurate moment localization. Note that this feature is also very different from previous approaches in terms of its sparse representation selection from several key elements. Thanks to the visual-language transformer backbone, the derived representation of each element contains enough context information from both modalities, sparsely selection does not decrease the representation ability for the moment candidate, but lets the model precisely catch the important elements for accurate moment localization.

We apply the L1 loss for the boundary regression:

$$L_{regress} = \frac{1}{Q}\sum_{i=1}^{Q}\left|{r_s^i}' - \left(t_s - {t_s^i}'\right)\right| + \left|{r_e^i}' - \left(t_e - {t_e^i}'\right)\right| \quad (6)$$

where Q is the number of moment candidates whose starting and ending ground truth scores are both greater than a threshold τ, (t_s, t_e) is the boundary of ground truth.

For the matching score, we adopt a truncated IoU value as the supervision signal. Specifically, we first compute the IoU score y between the regressed moment candidate, i.e. $({t_s}' + {r_s}', {t_e}' + {r_e}')$, and the ground truth moment, i.e. (t_s, t_e). Then, the IoU score y is truncated to 1 if it is not smaller than a threshold t_{max} and 0 if it is not larger than a threshold t_{min}. While other values of y keep the same. Then we also adopt the weighted binary cross entropy loss for the matching score, which is illustrated as follows:

$$L_{match} = \sum_{i=1}^{Z}\left(-y_i \, log \, {a_i}' - (1 - y_i) \, log \, (1 - {a_i}')\right) \cdot (y_i - {a_i}')^2 \quad (7)$$

where Z is the number of all moment candidates, ${a_i}'$ is the predicted matching score. Different from previous works computing the IoU score without regression, our IoU score is calculated between the regressed moment candidate and the ground truth moment, which makes our model can measure the quality of boundary regression.

To generate moment candidates, any moment proposal methods can be applied in our framework. For convenience, we follow the same candidate generation process as in [45]. Specifically, we first enumerate all possible segments which consist of any consecutive clips. Then, for short length segments, we densely pick them as moment candidates. For longer length segments, we gradually increase the sampling interval to sparsely select them as moment candidates. The key idea behind this is to remove the redundant segments which have large overlaps with the selected candidates. More details can be seen in [45].

3.4. Training and Inferring

During training, we pick the video-query pairs as input of our network. Similar to the original BERT [6], each word in the sentence sequence is randomly masked at a probability of 15%. For the masked word, its token is replaced with a special token of “[MASK]”. Then we let the model predict the masked words based on the unmasked words and the information from the visual sequence. Note that predicting some important words, e.g. nouns for objects and verbs for actions, needs the information from the video sequence. Thus, masked word prediction not only makes the transformer learn language dependencies but also better align the video and language modalities. The loss function for masked word prediction is the standard cross entropy loss. Then, this loss and the previous three losses for moment localization are summed together to train the whole network. For a fair comparison, we did not pretrain our transformer backbone on any other dataset. All the parameters are randomly initialized.

At inferring stage, our model takes the video sequence and sentence query without masking words as input and outputs the matching score and new boundary for each candidate. We rank the candidates according to their matching scores from high to low. Then we use NMS (Non Maximum Suppression) to remove the largely overlapped candidates and return the top 1 or top 5 candidates as the localized moments. Note that in order to keep the input consistency for query sentence during training and inferring, we can also input the query sentence without masked words at a small probability, e.g. 20%, in the training process.

4. Experiments

4.1. Dataset

ActivityNet Captions [15] contains 20K videos with 100K queries. The video average duration is 2 minutes. The videos in ActivityNet Captions contain diverse contents. We use the validation subset ”val_1” as our validation set and validation subset ”val_2” as our testing set. In our setting, there are 37,417, 17,505 and 17,031 moment-sentence pairs for training, validation and testing respectively.

TACoS [27] is widely used on the video grounding task. It contains 127 videos about cooking activities with an average duration of 7 minutes. TACoS is a very challenging dataset. The query sentences in TACoS contains multi-level activities with variable level of details. We follow the standard split [7], which includes 10,146, 4,589, and 4,083 moment-sentence pairs for training, validation and testing.

4.2. Experimental Settings

Evaluation Metric. Following previous works [7, 45], we utilize Rank n @ IoU=m to evaluate our method. It represents the percentage of correct localizations, where a correct localization is defined as there is at least one matched moment in the top-n generated moments. If the IoU between the generated moment and the ground truth moment is larger than m, the generated moment is matched. We set $n \in \{1, 5\}$ and $m \in \{0.5, 0.7\}$ for ActivityNet Captions, $n \in \{1, 5\}$ and $m \in \{0.3, 0.5\}$ for TACoS.

Implementation Details. We use AdamW [14, 22] to optimize our network. The batch size is set to 16 and the learning rate is 1×10^{-4}. The number of transformer layers is set to 6 and the feature dimension of all layers is set to 512. The number of heads is set to 16 and 32 for ActivityNet Captions and TACoS respectively. Following previous works, we utilize C3D network [33] to extract the video feature. We set the length of video clips N to 32 for ActivityNet Captions and 128 for TACoS. For both datasets, the standard deviation scalar α_s, α_m are set to $0.25, 0.21$ respectively. The

Table 1. Comparisons with state-of-the-arts on ActivityNet Captions dataset. All methods are based on the C3D video feature.

Method	Rank 1 IoU = 0.5	Rank 1 IoU = 0.7	Rank 5 IoU = 0.5	Rank 5 IoU = 0.7
MCN [1]	21.36	6.43	53.23	29.70
CTRL [7]	29.01	10.34	59.17	37.54
TGN [4]	27.93	-	44.20	-
ACRN [20]	31.67	11.25	60.34	38.57
CMIN [46]	43.40	23.88	67.95	50.73
QSPN [39]	33.26	13.43	62.39	40.78
ABLR [41]	36.79	-	-	-
TripNet [11]	32.19	13.93	-	-
SCDM [40]	36.75	19.86	64.99	41.53
DRN [43]	45.45	24.36	77.97	50.30
2D-TAN [45]	44.51	26.54	77.13	61.96
LGI [25]	41.51	23.07	-	-
DPIN [35]	47.27	28.31	77.45	60.03
CSMGAN [19]	**49.11**	29.15	77.43	59.63
FIAN [26]	47.90	29.81	77.64	59.66
Ours	48.02	**31.78**	**78.02**	**63.18**

Table 2. Comparisons with state-of-the-arts on TACoS dataset. All methods are based on the C3D video feature.

Method	Rank 1 IoU = 0.3	Rank 1 IoU = 0.5	Rank 5 IoU = 0.3	Rank 5 IoU = 0.5
CTRL [7]	18.32	13.30	36.69	25.42
ACRN [20]	19.52	14.62	34.97	24.88
ROLE [21]	15.38	9.94	31.17	20.13
VAL [29]	19.76	14.74	38.55	26.52
ACL-K [8]	24.17	20.01	42.15	30.66
CMIN [46]	24.64	18.05	38.46	27.02
QSPN [39]	20.15	15.23	36.72	25.30
SLTA [13]	17.07	11.92	32.90	20.86
ABLR [41]	19.50	9.40	-	-
DEBUG [23]	23.45	-	-	-
TripNet [11]	23.95	19.17	-	-
MCF [37]	18.64	12.53	37.13	24.73
TGN [4]	21.77	18.90	39.06	31.02
SCDM [40]	26.11	21.17	40.16	32.18
DRN [43]	-	23.17	-	33.36
2D-TAN [45]	37.29	25.32	57.81	45.04
CSMGAN [19]	33.90	27.09	53.98	41.22
FIAN [26]	33.87	28.58	47.76	39.16
DPIN [35]	46.74	32.92	62.16	50.26
Ours	**48.79**	**37.57**	**67.63**	**57.91**

threshold τ is set to 0.4, and the thresholds t_{min}, t_{max} for the IoU truncation are set to $0.5, 1.0$ respectively.

4.3. Comparison with State-of-the-Art Methods

We compare our proposed multi-stage aggregated transformer network with extensive state-of-the-art methods. The comparison results on ActivityNet Captions and TACoS are shown in Table 1-2. We can see that our method achieves significant improvements compared with all other methods, eapecically when the localization criterion is more rigorous. Although our method achieves Rank1@IoU=0.5 1.09 point lower than CSMGAN [19] on ActivityNet Captions, it outperforms CSMGAN in terms of all other metrics. Especially for Rank1@IoU=0.7 and Rank5@IoU=0.7 metrics, our method outperforms CSMGAN by 2.63 points and 3.55 points respectively. Note that IoU=0.7 is a more rigorous criterion to determine whether a localized moment is correct or not. This shows our method can localize the moment with higher quality. Besides, our method greatly outperforms CSMGAN on TACoS dataset by more than 10 points across all metrics. These results show the superiority of our method. We can also see that our improvements on TACoS are higher than the improvements on ActivityNet Captions. This is because the query sentences in TACoS contains multi-level activities with variable level of details. They are more challenging to be accurately localized. Taking the query sentence "woman slices the second kiwi and places it on the plate" as an example, it needs exact semantic alignment and stage-wise matching. In our model, each video clip can interact with each word in the query and our moment representation is stage sensitive, thus it greatly improves the performance.

Next, let us compare our model with other methods in more detail. Firstly, we compare our model with previous sliding window based methods: MCN [1], CTRL [7], ACRN [20], VAL [29] and ACL-K [8]. Those methods first use the sliding window to generate moment candidates, then directly fuse with the sentence query representation. They do not explore the detailed interactions between video and sentence. While QSPN [39], ABLR [41], CMIN [46], TGN [4] and SCDM [40] attempt to conduct cross interactions by using the sentence representation to attend the video contents or learn a sentence conditioned representation. However, they still independently encode the sentence query into a single vector, which inevitably losses some detailed semantics. Recent works, such as LGI [25], CSMGAN [19] and FIAN [26], explore the local and more detailed interactions between video and sentence to improve the performance. However, they neglect the different stages in the moment. While our proposed moment representation can match different stages, thus facilitates accurate localization. Besides, we conduct the fine-grained visual language alignments using our transformer backbone, which is more efficient and general than those well designed models.

Moreover, we compare our model with other typical methods, i.e., 2D-TAN [45], DPIN [35] and DRN [43]. In 2D-TAN, it arranges the features of all moment candidates into a 2D feature map. The candidate localizing at the (i, j) position in the map represents it starts at i-th clip and ends at j-th clip. Then, 2D-TAN utilizes 2D CNN to compute the matching score. Although it can learn the differences between adjacent candidates, the candidates must satisfy a fixed structure, which limits its adaptation. In order to improve the boundary localization accuracy, DPIN introduces two pathways which one for boundary prediction and one for semantic alignment. While our proposed multi-stage aggregated moment representation contains both semantics and stage/boundary information itself, and can be easily applied in other models as well. Different from other methods, DRN directly predicts the location regression from a video clip to the boundary. However, the queried moments have various semantics and temporal durations, regression from a clip is not very effective.

4.4. Ablation Study

In this section, we take the ablation study of investigating the effects of different factors in our proposed method. Specifically, we study the following variants of our model:

- **Conv-LSTM-Conv** utilizes LSTM to encode the query sentence to a vector and fuses it with the video sequence. After that, the fused sequence is modeled by multiple layers of temporal convolution. On top of the last convolution layer, we use stacked convolution [44] to obtain the representation for the moment candidate.

- **Conv-LSTM-MSA** is similar to Conv-LSTM-Conv, but it applies our proposed multi stage aggregation module to obtain the moment representation.

- **VLTrans-Mean/Conv/RoI/MSA** encodes the visual and language sequences into a unified sequence. Then, the unified sequence is feed-forwarded to a single standard BERT to model visual and language interactions. **Mean**, **Conv**, **RoI**, **MSA** represent using the mean pooling, stacked convolution, RoI pooling and our multi stage aggregation module to obtain the moment representation respectively.

- **De-VLTrans-MSA** is similar to VLTrans-MSA, but utilizes the decoupled visual language transformer instead of the standard single BERT to model visual and language interactions.

- **TBERT-MSA** is inspired by [32]. It uses two BERT streams to process visual and language sequence separately. The two BERT streams introduce additional cross-modal layers to achieve multi-modality interactions. Then it also applies our multi stage aggregation module to obtain the moment representation.

- **VLTrans-MSA-4X** based models are similar to VLTrans-MSA, but keep VLTrans-MSA with the same number of parameters as De-VLTrans-MSA. **V1** and **V2** represent 4 times of parameters in width and depth respectively, and **V3** represents 2 times of parameters in width with 2 times of parameters in depth.

- **De-VLTrans-MSA-2S/4S** keeps the same architecture as De-VLTrans-MSA, but uses the different number of moment stages. **2S** represents keeping only the starting and ending stages, while **4S** represents keeping the starting, left center, right center and ending stages.

- **De-VLTrans-MSA-UM** is same as De-VLTrans-MSA except that it is trained without the masked words prediction.

The experimental results of these variants are shown in Table 3, in which we can obtain the following observations:

Does transformer help? Comparing Conv-LSTM-Conv with VLTrans-Conv, and Comparing Conv-LSTM-MSA with VLTrans-MSA, De-VLTrans-MSA, TBERT-MSA respectively, we can see that using the transformer backbone can significantly improve the performance. This is because the transformer backbone enables dynamical and fine-grained visual language interaction and alignment, and thus improves the localization performance.

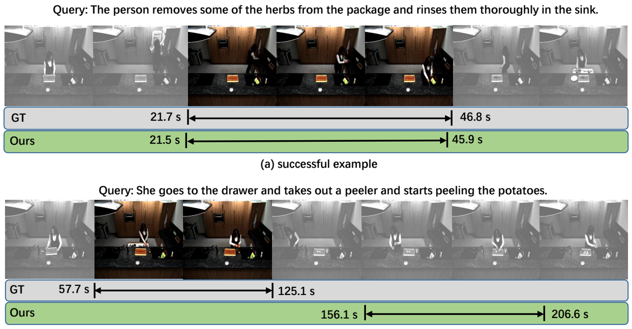
Does decoupling parameters perform better? When decoupling the parameters for visual and language modalities, De-VLTrans-MSA achieves better performance than VLTrans-MSA. This verifies our consideration that different modalities have modality specific contents and relation patterns, we should decouple the parameters for better modelling different modalities. We can also see our proposed De-VLTrans-MSA outperforms the VLTrans-MSA-4X based models. In fact, the VLTrans-MSA-4X based models even achieve worse performance than VLTrans-MSA. This shows simply adding the number of parameters of VLTrans-MSA cannot increase the performance, and demonstrates the effectiveness of our De-VLTrans-MSA as well. We also compare our De-VLTrans-MSA with TBERT-MSA that uses two BERT streams to process visual and language sequence separately. Generally, we can see our De-VLTrans-MSA achieves better performance than TBERT-MSA. The reason is that our De-VLTrans-MSA is more architecture compact and computation efficient.

How is the effect of multi-stage aggregation module? In both Conv-LSTM based models and VLTrans based models, topping our multi-stage aggregation module achieves significant improvements than other ways, e.g. stacked convolution, mean pooling and RoI pooling. Because our proposed representation captures the stage specific information, it can match the moment more accurately. Our multi-stage aggregated module can easily replace other ways on different models. It is a universal method for candidate feature extraction. Note that the mean pooling operation performs significantly worse than other methods on TACoS dataset. This is due to that the mean pooling operation totally losses the stage-specific information. While the activity length in TACoS is more variable, lacking this information has a great influence.

How does the number of moment stages influence performance? Overall, keeping more stages, i.e. De-VLTrans-MSA-4S, does not improve the performance. While keeping only two stages, i.e. De-VLTrans-MSA-2S, also decrease the performance. For us, it is intuitive to divide an event into three stages, i.e. starting, middle and ending

Table 3. Ablation study on ActivityNet Captions and TACoS datasets. All methods are based on the C3D video feature.

Method	ActivityNet Captions				TACoS			
	Rank 1 IoU = 0.5	Rank 1 IoU = 0.7	Rank 5 IoU = 0.5	Rank 5 IoU = 0.7	Rank 1 IoU = 0.3	Rank 1 IoU = 0.5	Rank 5 IoU = 0.3	Rank 5 IoU = 0.5
Conv-LSTM-Conv	41.12	24.23	76.31	55.89	35.77	27.27	60.26	48.09
Conv-LSTM-MSA	45.16	27.81	76.70	58.65	40.66	31.39	63.86	50.51
VLTrans-Mean	45.92	27.28	77.34	61.61	44.99	30.92	66.31	51.96
VLTrans-Conv	45.36	28.41	77.68	60.81	46.16	35.02	**68.41**	55.56
VLTrans-RoI	45.02	28.42	77.82	60.40	45.11	34.92	66.06	55.64
VLTrans-MSA	46.96	30.27	77.69	62.28	47.26	36.89	66.41	57.09
De-VLTrans-MSA	**48.02**	**31.78**	78.02	63.18	**48.79**	37.57	67.63	**57.91**
TBERT-MSA	46.74	30.12	76.82	61.32	48.61	**37.94**	64.66	55.24
VLTrans-MSA-4X-V1	45.52	28.74	77.48	62.19	45.24	35.04	66.53	56.94
VLTrans-MSA-4X-V2	20.74	12.20	57.34	38.95	6.70	2.10	29.09	14.15
VLTrans-MSA-4X-V3	45.94	27.94	78.45	62.59	45.19	36.14	65.81	55.51
De-VLTrans-MSA-2S	46.84	30.15	78.00	62.66	44.56	34.12	63.53	53.59
De-VLTrans-MSA-4S	47.69	31.24	**78.51**	**63.41**	47.91	36.69	67.61	57.34
De-VLTrans-MSA-UM	46.26	28.82	77.36	61.81	45.64	34.79	65.31	55.91


Query: The person removes some of the herbs from the package and rinses them thoroughly in the sink.
GT
21.7 s
46.8 s
Ours
21.5 s
45.9 s
Query: She goes to the drawer and takes out a peeler and starts peeling the potatoes.
GT
57.7 s
125.1 s
Ours
156.1 s
206.6 s

(a) successful example

(b) failed example

Figure 2. The detected examples of our model on TACoS. **GT** is the ground truth moment, **Ours** is the result of De-VLTrans-MSA.

stages. Too few stages will miss some important contents. However, adding more stages is not always necessary and may even cause redundancy. Because we impose a layer to predict stage scores for each clip, the clip representations have been pushed to fuse useful surrounding contents.

How useful is the masked language modeling? When training without the masked words prediction, De-VLTrans-MSA-UM achieves significantly worse performance. As stated before, masked word prediction not only makes the transformer learn language dependencies but also better align the video and language modalities.

4.5. Qualitative Results

Figure 2 visualizes some detected examples of our model on TACoS. From Figure 2 (a), we can see our model localizes the moment very accurately. This demonstrates the effectiveness of our model. In Figure 2 (b), our model fails to localize the correct moment. The reason is that "peeling the potatoes" and "cutting the potatoes" have similar contents thus are easily confused. This inspires us that accurate visual semantics learning is very important. In the future, we will attempt to use large-scale video-language data on the internet, e.g. the videos and their captions on YouTube, to pretrain our model.

5. Conclusion

This paper presents a novel multi-stage aggregated transformer network for temporal language localization in videos. Specifically, we introduce a new visual-language transformer backbone, which enables dependency modeling among all the input elements from both visual and language sequences to effectively model the fine-grained visual-language alignment. Our proposed backbone keeps both structure unified and modality specific. Furthermore, we also propose a multi-stage aggregation module topped on the transformer backbone, in which we compute three stage-specific representations corresponding to different temporal stages and conduct the multi-stage aggregation to obtain the more discriminative feature for accurate moment localization. These two components are deeply integrated to form an efficient and effective network. Our proposed model also has good scalability. We can use a large amount of video-language data to pre-train our network. We believe our work will promote the future research of this new kind of architecture for temporal language localization.

Acknowledgement This work was supported in part by National Natural Science Foundation of China under grant U20B2063, the National Key Research and Development Program of China under grant No. 2018AAA0102200, the Sichuan Science and Technology Program, China, under grant 2020YFS0057, and Dongguan Songshan Lake Introduction Program of Leading Innovative and Entrepreneurial Talents.

References

[1] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *Proceedings of the IEEE international conference on computer vision*, pages 5803–5812, 2017. 1, 2, 3, 6

[2] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5297–5307, 2016.

[3] Yu-Wei Chao, Sudheendra Vijayanarasimhan, Bryan Seybold, David A Ross, Jia Deng, and Rahul Sukthankar. Rethinking the faster r-cnn architecture for temporal action localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1130–1139, 2018. 1

[4] Jingyuan Chen, Xinpeng Chen, Lin Ma, Zequn Jie, and Tat-Seng Chua. Temporally grounding natural sentence in video. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 162–171, 2018. 2, 6

[5] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *European Conference on Computer Vision*, pages 104–120. Springer, 2020. 2, 3, 4

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. 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)*, pages 4171–4186, 2019. 5

[7] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. Tall: Temporal activity localization via language query. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5267–5275, 2017. 1, 2, 3, 5, 6

[8] Runzhou Ge, Jiyang Gao, Kan Chen, and Ram Nevatia. Mac: Mining activity concepts for language-based temporal localization. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 245–253. IEEE, 2019. 3, 6

[9] Soham Ghosh, Anuva Agarwal, Zarana Parekh, and Alexander Hauptmann. Excl: Extractive clip localization using natural language descriptions. *arXiv preprint arXiv:1904.02755*, 2019. 2

[10] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015. 1

[11] Meera Hahn, Asim Kadav, James M Rehg, and Hans Peter Graf. Tripping through time: Efficient localization of activities in videos. *arXiv preprint arXiv:1904.09936*, 2019. 6

[12] Dongliang He, Xiang Zhao, Jizhou Huang, Fu Li, Xiao Liu, and Shilei Wen. Read, watch, and move: Reinforcement learning for temporally grounding natural language descriptions in videos. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8393–8400, 2019. 3

[13] Bin Jiang, Xin Huang, Chao Yang, and Junsong Yuan. Cross-modal video moment retrieval with spatial and language-temporal attention. In *Proceedings of the 2019 on International Conference on Multimedia Retrieval*, pages 217–225, 2019. 6

[14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[15] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *Proceedings of the IEEE international conference on computer vision*, pages 706–715, 2017. 5

[16] Gen Li, Nan Duan, Yuejian Fang, Ming Gong, Daxin Jiang, and Ming Zhou. Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training. In *AAAI*, pages 11336–11344, 2020. 2, 3, 4

[17] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. In *The 58th Annual Meeting of the Association for Computational Linguistics*, 2020. 2, 3, 4

[18] Xin Li, Tianwei Lin, Xiao Liu, Wangmeng Zuo, Chao Li, Xiang Long, Dongliang He, Fu Li, Shilei Wen, and Chuang Gan. Deep concept-wise temporal convolutional networks for action localization. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 4004–4012, 2020. 1

[19] Daizong Liu, Xiaoye Qu, Xiao-Yang Liu, Jianfeng Dong, Pan Zhou, and Zichuan Xu. Jointly cross-and self-modal graph attention network for query-based moment localization. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 4070–4078, 2020. 1, 3, 6

[20] Meng Liu, Xiang Wang, Liqiang Nie, Xiangnan He, Baoquan Chen, and Tat-Seng Chua. Attentive moment retrieval in videos. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 15–24, 2018. 1, 3, 6

[21] Meng Liu, Xiang Wang, Liqiang Nie, Qi Tian, Baoquan Chen, and Tat-Seng Chua. Cross-modal moment localization in videos. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 843–851, 2018. 3, 6

[22] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 5

[23] Chujie Lu, Long Chen, Chilie Tan, Xiaolin Li, and Jun 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)*, pages 5147–5156, 2019. 6

[24] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Advances in Neural Information Processing Systems*, pages 13–23, 2019. 4

[25] Jonghwan Mun, Minsu Cho, and Bohyung Han. Local-global video-text interactions for temporal grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vi-*

sion and Pattern Recognition, pages 10810–10819, 2020. 1, 3, 6

[26] Xiaoye Qu, Pengwei Tang, Zhikang Zou, Yu Cheng, Jianfeng Dong, Pan Zhou, and Zichuan Xu. Fine-grained iterative attention network for temporal language localization in videos. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 4280–4288, 2020. 1, 3, 6

[27] Michaela Regneri, Marcus Rohrbach, Dominikus Wetzel, Stefan Thater, Bernt Schiele, and Manfred Pinkal. Grounding action descriptions in videos. *Transactions of the Association for Computational Linguistics*, 1:25–36, 2013. 5

[28] Zheng Shou, Jonathan Chan, Alireza Zareian, Kazuyuki Miyazawa, and Shih-Fu Chang. Cdc: Convolutional-deconvolutional networks for precise temporal action localization in untrimmed videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5734–5743, 2017. 1

[29] Xiaomeng Song and Yahong Han. Val: Visual-attention action localizer. In *Pacific Rim Conference on Multimedia*, pages 340–350. Springer, 2018. 6

[30] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vl-bert: Pre-training of generic visual-linguistic representations. In *International Conference on Learning Representations*, 2020. 2, 3, 4

[31] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7464–7473, 2019. 2, 3, 4

[32] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*, 2019. 4, 7

[33] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015. 5

[34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 4

[35] Hao Wang, Zheng-Jun Zha, Xuejin Chen, Zhiwei Xiong, and Jiebo Luo. Dual path interaction network for video moment localization. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 4116–4124, 2020. 1, 6

[36] Weining Wang, Yan Huang, and Liang Wang. Language-driven temporal activity localization: A semantic matching reinforcement learning model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 334–343, 2019. 3

[37] Aming Wu and Yahong Han. Multi-modal circulant fusion for video-to-language and backward. In *IJCAI*, volume 3, page 8, 2018. 6

[38] Huijuan Xu, Abir Das, and Kate Saenko. R-c3d: Region convolutional 3d network for temporal activity detection. In *Proceedings of the IEEE international conference on computer vision*, pages 5783–5792, 2017. 1

[39] Huijuan Xu, Kun He, Bryan A Plummer, Leonid Sigal, Stan Sclaroff, and Kate Saenko. Multilevel language and vision integration for text-to-clip retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9062–9069, 2019. 3, 6

[40] Yitian Yuan, Lin Ma, Jingwen Wang, Wei Liu, and Wenwu Zhu. Semantic conditioned dynamic modulation for temporal sentence grounding in videos. In *Advances in Neural Information Processing Systems*, pages 536–546, 2019. 1, 6

[41] Yitian Yuan, Tao Mei, and Wenwu 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*, volume 33, pages 9159–9166, 2019. 1, 2, 6

[42] Runhao Zeng, Wenbing Huang, Mingkui Tan, Yu Rong, Peilin Zhao, Junzhou Huang, and Chuang Gan. Graph convolutional networks for temporal action localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7094–7103, 2019. 1

[43] Runhao Zeng, Haoming Xu, Wenbing Huang, Peihao Chen, Mingkui Tan, and Chuang Gan. Dense regression network for video grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10287–10296, 2020. 1, 2, 6

[44] Da Zhang, Xiyang Dai, Xin Wang, Yuan-Fang Wang, and Larry S Davis. Man: Moment alignment network for natural language moment retrieval via iterative graph adjustment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1247–1257, 2019. 3, 7

[45] Songyang Zhang, Houwen Peng, Jianlong Fu, and Jiebo Luo. Learning 2d temporal adjacent networks formoment localization with natural language. In *AAAI*, 2020. 1, 3, 5, 6

[46] Zhu Zhang, Zhijie Lin, Zhou Zhao, and Zhenxin Xiao. Cross-modal interaction networks for query-based moment retrieval in videos. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 655–664, 2019. 3, 6

[47] Peisen Zhao, Lingxi Xie, Chen Ju, Ya Zhang, Yanfeng Wang, and Qi Tian. Bottom-up temporal action localization with mutual regularization. In *European Conference on Computer Vision*, pages 539–555. Springer, 2020. 1

[48] Yue Zhao, Yuanjun Xiong, Limin Wang, Zhirong Wu, Xiaoou Tang, and Dahua Lin. Temporal action detection with structured segment networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2914–2923, 2017. 1