Sketch, Ground, and Refine: Top-Down Dense Video Captioning

Chaorui Deng[*1,3], Shizhe Chen[**2], Da Chen[*3], Yuan He[3], Qi Wu[†1]
[1]University of Adelaide, [2]INRIA, [3]Alibaba Group
{chaorui.deng,qi.wu01}@adelaide.edu.au, cshizhe@gmail.com, {chen.cd,heyuan.hy}@alibaba-inc.com

Abstract

The dense video captioning task aims to detect and describe a sequence of events in a video for detailed and coherent storytelling. Previous works mainly adopt a "detect-then-describe" framework, which firstly detects event proposals in the video and then generates descriptions for the detected events. However, the definitions of events are diverse which could be as simple as a single action or as complex as a set of events, depending on different semantic contexts. Therefore, directly detecting events based on video information is ill-defined and hurts the coherency and accuracy of generated dense captions. In this work, we reverse the predominant "detect-then-describe" fashion, proposing a top-down way to first generate paragraphs from a global view and then ground each event description to a video segment for detailed refinement. It is formulated as a Sketch, Ground, and Refine process (SGR). The sketch stage first generates a coarse-grained multi-sentence paragraph to describe the whole video, where each sentence is treated as an event and gets localised in the grounding stage. In the refining stage, we improve captioning quality via refinement-enhanced training and dual-path cross attention on both coarse-grained event captions and aligned event segments. The updated event caption can further adjust its segment boundaries. Our SGR model outperforms state-of-the-art methods on ActivityNet Captioning benchmark under traditional and story-oriented dense caption evaluations. Code will be released at github.com/bearcatt/SGR.

1. Introduction

Video understanding is an important research topic in computer vision. Thanks to the development of deep learning and large-scale datasets [3, 10], recent video understanding approaches have achieved promising performance in action recognition [4, 5, 35] and temporal action localisation [15, 25, 41]. However, an obvious limitation of these approaches is that their predictions are based on a pre-defined discrete set of action categories, which lack many fine-grained details of the video information. In order to provide more detailed information in the video, the task of Dense Video Captioning (DVC) [11] aims to discover a sequence of key events in a video and describe them using a coherent story. Therefore, DVC can benefit many real-world applications such as content-based video retrieval and recommendation, and has become an important task in language-based video understanding research.

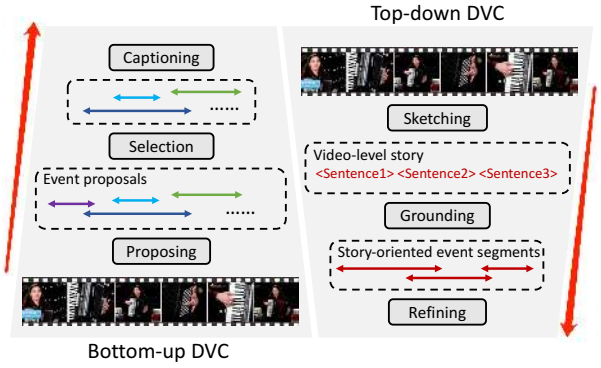

Figure 1. Comparisons between top-down and bottom-up DVC frameworks. Bottom-up DVC breaks the DVC task into multiple "event captioning" sub-tasks with an event detection process. While our SGR captions the whole video in a unified fashion and localises each event afterwards.

DVC is a challenging task since the generated event captions are expected to be correct, concise, and coherent in the context of a video-level story, so as to support video understanding. Most existing methods [11, 13, 33, 43] in this field adopt a common "detect-then-describe" framework which solves the problem from a straightforward bottom-up perspective, *i.e.*, first detect a large set of event proposals (even up to 10^3), then caption each proposal to obtain dense descriptions. However, one obvious issue of this framework is that the event proposals are independently generated without considering their temporal correlations, thus making the resultant event captions highly redundant and incoherent.

To address this issue, Mun *et al*. [18] proposed a streamlined DVC framework that learns a proposal selection mod-

[*]Equal contribution, [†]Corresponding author
[*]Work done while at the Renmin University of China.

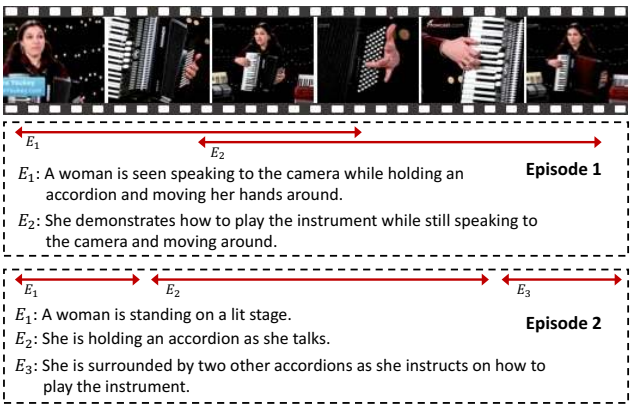

Figure 2. Illustration of the diverse definition of event proposals. The first episode consists of salient events, which are long and informative, while in the second episode, a simple action is also considered as an event, which is very short and indistinctive.

ule to find a small subset of the event proposals that occur sequentially in the video. In this way, they explicitly consider the temporal correlations among the selected event proposals. However, their framework is still limited due to the *ill-defined event proposal generation*. As shown in Figure 2, the definition of events varies dramatically in practice, from as simple as an action (E_1, Episode 2) to as complicated as a combination of multiple salient events (E_1, Episode 1). Therefore, detecting the event proposals based solely on the video information like [18] and other bottom-up DVC approaches may have an ill-defined target, *i.e*., the detector is not aware of which kinds of event proposals are suitable to be captioned. As a result, the quality of the generated event captions may be negatively affected.

In this paper, we reverse the predominant "detect-then-describe" fashion and propose to solve DVC from a top-down perspective, *i.e*., generating a video-level story at first and then ground each sentence in the story to video segment for detailed refinement. By doing this, the event segments are predicted not only based on the visual information, but also the semantic coherence from the text. Thus the aforementioned issues are waived. Our model is formulated as a Sketch, Ground, and Refine (SGR) procedure.

In the **Sketch** stage, we focus on the *structure coherency* of the event captions. We first leverage the entire video information to generate a coherent video-level paragraph so as to describe the video from a global perspective. Note that, the sentences in this paragraph naturally defines a sequence of story-oriented events in the video. Therefore, it is not necessary for our framework to generate a large number of event proposals as in bottom-up approaches. In fact, our top-down SGR contains no event proposal generation process. Instead, we use a video **Grounding** module to localise the sentences (*i.e*., events) in our video-level paragraph.

The sentences generated so far may lack some fine-grained details since we did not explicitly consider their event-specific information. Thus, in the **Refine** stage, we focus on the *fine-grained details* of the event captions. Specifically, we propose a Dual-Path Cross Attention module to dynamically focus on the event-level information and the coarse-grained sentence. A refinement-enhanced training scheme is also designed to facilitate the refinement. Based on the refined descriptions, we further adjust their event segments by feeding them into the grounding module again. We illustrate the difference between our SGR and the previous DVC frameworks in Figure 1.

We evaluate our method on the benchmark ActivityNet Captioning dataset [11] and YouCook2 dataset [42]. Our model achieves state-of-the-art performance under both traditional and story-oriented dense caption evaluations.

2. Related Works

Video Captioning aims to describe a video using a single sentence. Recent approaches mostly follow the encoder-decoder framework [19, 20, 21, 31, 32, 34], where the encoder is typically a Convolutional Neural Networks (CNN) followed by a Recurrent Neural Network (RNN), and the decoder is an RNN that predict one token at each decoding step. However, in practice, a video is generally informative and contains multiple events, which is beyond the capacity of a short sentence. Therefore, some recent works [12, 36, 38] appealed to generating a long paragraph to describe the video in detail, where each sentence in the paragraph may focus on a specific event in the video. On top of this, Dense Video Captioning further requires to localise the temporal boundaries of these events.

Dense Video Captioning (DVC) is first proposed by Krishna *et al*. [11], where they combine an event proposal module and a video captioning module to tackle the DVC task: the proposal module first selects a large set of event segments from the video, then the captioning module captions each event segment, *i.e*., the detect-then-describe framework. Recent works [13, 33, 43] improved this framework by making the event proposal module and the video captioning module end-to-end trainable, so that the annotated event captions can also guide the training of the event proposal module. Specifically, Li *et al*. [13] propose to predict a "descriptiveness" score for each event proposal during the proposal generation process, which measures the complexity of describing each proposal while also guides the proposal generation. Wang *et al*. [33] utilise a bidirectional video encoder to exploits both past and future contexts to make proposal predictions, and further adopt this bidirectional representation for event captioning. Zhou *et al*. [43] try to bridge the event detection and caption module into a unified model by applying a differentiable masking mechanism over a Transformer [29] based encoder-decoder archi-

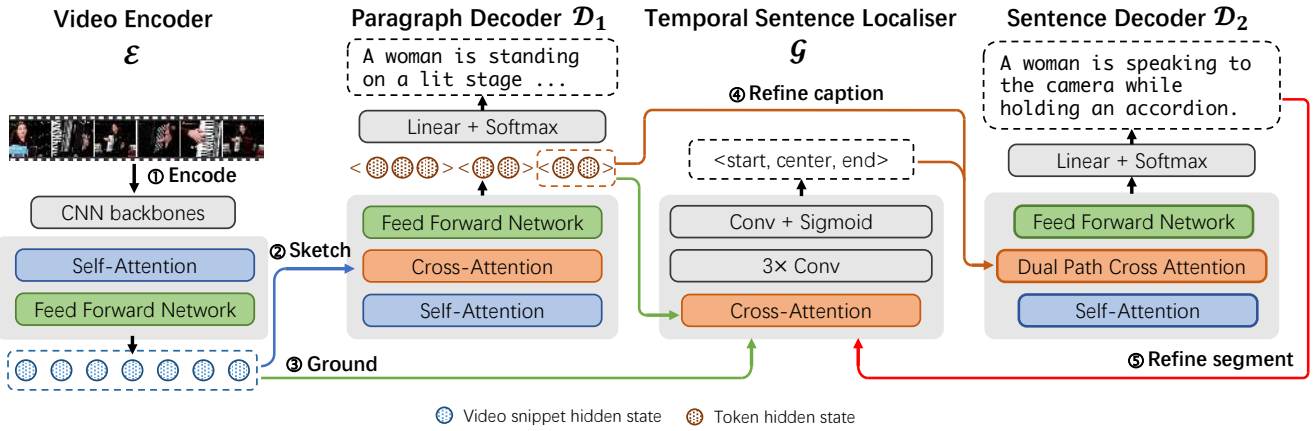

Figure 3. The overall framework of the proposed SGR. The video encoder $\mathcal{E}$ first encodes the video into a feature sequence. Based on it, the paragraph decoder $\mathcal{D}_1$ then sketches a video-level story describing the whole video. The localiser $\mathcal{G}$ then ground the sentences in the story to the video segments. Then, the sentence decoder $\mathcal{D}_2$ refines the sentences by leveraging their event-specific information, while the refined sentences are further used to adjust the segment boundaries.

tecture. As discussed in the introduction, the above methods tend to generate a large number of independent proposals for dense captioning which may lead to redundant or inconsistent results. To tackle this issue, Mun *et al*. [18] apply an event sequential generation module to find a small subset of event proposal that occurs sequentially in the video, so as to reduce the number of proposal and make the caption results more coherent.

The aforementioned DVC approaches all follow the bottom-up "detect-then-describe" framework, which is problematic since the video-based event proposing is ill-defined. Unlike them, we propose a top-down DVC framework termed "Sketch, Ground, and Refine" (SGR), which contains no event proposing process. SGR shares some high-level ideas with works from other areas. In [7], the authors propose to perform image captioning using global image features while refining the captions using region features. [24] for text-video retrieval task retrieves video at paragraph-level while localizing segments at sentence-level. The paragraph-level retrieval result can be improved using sentence-level predictions.

3. Method

3.1. Overview

As illustrated in Figure 3, our framework consists of four modules: video encoder $\mathcal{E}$, temporal sentence localiser $\mathcal{G}$, coarse-grained paragraph decoder $\mathcal{D}_1$ and fine-grained sentence decoder $\mathcal{D}_2$. Compared with traditional bottom-up methods [11, 13, 18], the proposed SGR inverses the standard "detect-then-describe" pipeline for dense video captioning. Algorithm 1 presents how the framework generates dense video captions for a video V. Specifically, we first

Algorithm 1 Top-Down Dense Video Captioning

procedure DVC(V)
  $\boldsymbol{H}_v \leftarrow \mathcal{E}(V)$ ▷ encode
  $\{S_i\}_{i=1}^n \leftarrow \mathcal{D}_1(\boldsymbol{H}_v)$ ▷ sketch
  $\{E_i\}_{i=1}^n \leftarrow \mathcal{G}(\boldsymbol{H}_v, \{S_i\}_{i=1}^n)$ ▷ ground
  $\{S_i^r\}_{i=1}^n \leftarrow \mathcal{D}_2(\boldsymbol{H}_v, \{E_i\}_{i=1}^n, \{S_i\}_{i=1}^n)$ ▷ refine cap
  $\{E_i^r\}_{i=1}^n \leftarrow \mathcal{G}(\boldsymbol{H}_v, \{S_i^r\}_{i=1}^n)$ ▷ refine seg
  return $\{S_i^r\}_{i=1}^n, \{E_i^r\}_{i=1}^n$

convert V into a sequence of snippet representations $\boldsymbol{H}_v$ with the video encoder $\mathcal{E}$. Then, we sketch the structure of events in V by generating a paragraph P consisting of multiple sentences $\{S_i\}_{i=1}^n$ through decoder $\mathcal{D}_1$, and then localise the video segment E_i of each sentence S_i in P via the localiser $\mathcal{G}$. Afterwards, the event descriptions are refined with another decoder $\mathcal{D}_2$, given the coarse-grained S_i and its localised segment E_i from the last step. In the end, the video segment can again be adjusted to be more precise by $\mathcal{G}$ based on the refined event descriptions. Denote the refined S_i and E_i as S_i^r and E_i^r, respectively. Note that we reuse some features at different steps for computational efficiency, as shown in Figure 3.

We adopt a two-step training strategy to train the model. **Step 1**, we jointly train $\mathcal{E}$, $\mathcal{D}_1$ and $\mathcal{G}$ for paragraph sketching and temporal sentence grounding under the loss function $L = L_s + \lambda L_g$, where L_s is the paragraph generation loss and L_g is the temporal grounding loss. The hyper-parameter λ balances the two loss terms. **Step 2**, based on the trained $\mathcal{E}$, $\mathcal{D}_1$ and $\mathcal{G}$, we optimise $\mathcal{D}_2$ specifically with other modules fixed. $\mathcal{D}_2$ is trained by objective L_r through Reinforcement Learning with refine-specific rewards. We describe the details in the following.

3.2. Encoding: Contextual Video Encoder

Our video encoder $\mathcal{E}$ is composed of a fixed CNN backbone and a stack of transformer layers, which aims to encodes video V into a sequence of context-aware snippet representations $\boldsymbol{H}_v$.

The video is initially divided into T fixed-length snippets with duration of τ_v for each snippet. The CNN backbones independently extract features for each snippet as $\boldsymbol{F}_v \in \mathbb{R}^{T\times d_f}$, where d_f is the feature dimension. Since the temporal context is important for event understanding, we thus use transformer layers [29] with multi-head attention (MHA) to encode long-range dependencies among elements in $\boldsymbol{F}_v$. Let $\boldsymbol{H}_v^l \in \mathbb{R}^{T\times d_m}$ be the input to the l-th transformer encoding layer (d_m is the hidden dimension), the output $\boldsymbol{H}_v^{l+1}$ is computed as follows:

$$\boldsymbol{H}_v^{l+1} = \text{FFN}(\boldsymbol{H}_v^l + \text{MHA}(\boldsymbol{H}_v^l, \boldsymbol{H}_v^l, \boldsymbol{H}_v^l)) \tag{1}$$

where,

$$\begin{aligned}
&\text{FFN}(x) = \max(0, x\boldsymbol{W}_1 + \boldsymbol{b}_1)\boldsymbol{W}_2 + \boldsymbol{b}_2 \\
&\text{MHA}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{Cat}(\{\text{ATT}_i(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V})\}_{i=1}^h)\boldsymbol{W}_O \\
&\text{ATT}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \sigma(\frac{(\boldsymbol{Q}\boldsymbol{W}_Q)(\boldsymbol{K}\boldsymbol{W}_K)^T}{\sqrt{d_m}})(\boldsymbol{V}\boldsymbol{W}_V)
\end{aligned} \tag{2}$$

$\boldsymbol{H}_v^0$ is a linear transformation of $\boldsymbol{F}_v$ into dimension of d_m. $\text{Cat}(\cdot)$ denotes vector concatenation and $\boldsymbol{W}_*$ are learnable parameters. σ is the softmax function. We omit layer normalisation and residual connection for simplicity. The output from the final layer is $\boldsymbol{H}_v$.

3.3. Sketching: Paragraph Decoder

The goal of sketching is to organise a story-oriented event structure in the video. We employ a paragraph decoder $\mathcal{D}_1$ to this end. When $\mathcal{D}_1$ describes the video V from a global perspective, it naturally addresses the challenges of capturing diverse event categories and semantic relationships between different events.

We apply a transformer-based decoder as $\mathcal{D}_1$. Denote the hidden states of the l-th layer of $\mathcal{D}_1$ as $\boldsymbol{H}_p^l$. Based on the encoded video feature $\boldsymbol{H}_v$ and $\boldsymbol{H}_p^l$, the l-th layer generates semantic hidden states $\boldsymbol{H}_p^{l+1}$ as follows:

$$\boldsymbol{H}_p^{l,s} = \boldsymbol{H}_p^l + \text{MHA}(\boldsymbol{H}_p^l, \boldsymbol{H}_p^l, \boldsymbol{H}_p^l) \tag{3}$$

$$\boldsymbol{H}_p^{l,c} = \boldsymbol{H}_p^{l,s} + \text{MHA}(\boldsymbol{H}_p^{l,s}, \boldsymbol{H}_v, \boldsymbol{H}_v) \tag{4}$$

$$\boldsymbol{H}_p^{l+1} = \boldsymbol{H}_p^{l,c} + \text{FFN}(\boldsymbol{H}_p^{l,c}) \tag{5}$$

FFN and MHA are the same functions as defined in the Eq. 2. $\boldsymbol{H}_p^0$ is the input word embedding with position encoding. Denoting the final hidden states of $\mathcal{D}_1$ as $\boldsymbol{H}_p \in \mathbb{R}^{L_p\times d_m}$ (L_p is the length of the generated paragraph P), a captioning head maps $\boldsymbol{H}_p$ into word distributions, *i.e.*, the probability of predicting the j-th word w_j is $p(w_j|w_{<j}, \boldsymbol{H}_v) = \text{softmax}(\boldsymbol{H}_{p,j}\boldsymbol{W}_{emb})$, where $\boldsymbol{W}_{emb}$ is the word embedding matrix.

Training. We use "teacher forcing" scheme to train the paragraph generation modules. Given the ground-truth paragraph P^*, the objective is to minimise the negative log-likelihood of the ground-truth words w_j^* in P^* given all preceding ground-truth words $w_{<j}^*$ and the video feature sequence $\boldsymbol{H}_v$:

$$L_s = \sum_{w_j^*\in P^*} -\log p(w_j^*|w_{<j}^*, \boldsymbol{H}_v). \tag{6}$$

To address the exposure bias issue of teacher forcing, scheduled sampling [2] and label smoothing [27] are also applied during training.

Inference. Since the paragraph decoder is mainly used to sketch event structures in the video that are semantically coherent, we mainly focus on the semantic order and fluency. Therefore, we use greedy decoding to generate paragraph in inference for efficiency, and semantic hidden states $\boldsymbol{H}_p$ at the last decoding layer will be used afterwards.

3.4. Grounding: Temporal Sentence Localiser

Since we do not perform event proposal generation at the beginning of dense video captioning, a grounding module $\mathcal{G}$ is necessary to align the generated event description to its corresponding video segment.

Different from standard temporal sentence localisation task where the query description is a single sentence without any contexts, the query event description S_i in our setting also contains the context information from the paragraph $P = \{S_i\}_{i=1}^n$ for localisation. Such contexts not only provide semantics of neighbourhood events for the query event, but also inform about the relative order of the event, *i.e.* the event of last sentence S_n is more likely happens at the end of the video. Therefore, we propose a temporal sentence localiser $\mathcal{G}$ that employs paragraph context to benefit event description localisation.

For each sentence $S_i \in P$, we first extract its semantic representation $\boldsymbol{H}_{s_i} \in \mathbb{R}^{L_{s_i}\times d_m}$ which is a subset of $\boldsymbol{H}_p$ from $\mathcal{D}_1$. L_{s_i} is the length of S_i. As a multi-head attention (MHA) transformer is used to represent textual features, the $\boldsymbol{H}_{s_i}$ naturally incorporates event contexts from previously generated sentences, and does not require additional computations. Then, we dynamically merges the video feature sequence $\boldsymbol{H}_v$ with the query text $\boldsymbol{H}_{s_i}$ as follows:

$$\boldsymbol{H}_v^{s_i} = \boldsymbol{H}_v + \text{ATT}(\boldsymbol{H}_v, \boldsymbol{H}_{s_i}, \boldsymbol{H}_{s_i}) \tag{7}$$

We then apply several 1-D convolutions over temporal dimension on top of $\boldsymbol{H}_v^{s_i}$ in order to predict more accurate temporal boundaries over time. Denoting the output feature as $\tilde{\boldsymbol{H}}_v^{s_i}$, we use three linear classifiers to predict three confidence scores $\boldsymbol{c}_s, \boldsymbol{c}_c, \boldsymbol{c}_e \in \mathbb{R}^T$:

$$\boldsymbol{c}_x = f(\tilde{\boldsymbol{H}}_v^{s_i}\boldsymbol{W}_x), \text{ for } x \in \{s, c, e\} \tag{8}$$

where f is sigmoid function. The t-th values of $\boldsymbol{c}_s, \boldsymbol{c}_c, \boldsymbol{c}_e$ denote the confidences for the t-th snippet to be the start, centre and end location of the corresponding video segment E_i respectively.

Training. Suppose $E_i^* = [t_i^s, t_i^e]$ is the target temporal location for the ground-truth event description S_i^*, where t_i^s and t_i^e are the start and end timestamp respectively, we can obtain the ground-truth labels $\boldsymbol{c}_s^*$, $\boldsymbol{c}_c^*$, and $\boldsymbol{c}_e^*$ for snippets in V, indicating whether a snippet is inside the start, centre and end regions of the target event, following recent works of temporal action localisation [14, 15].

Specifically, we denote the start region of E_i^* as $r_i^s = [t_i^s - \alpha_1 d_i, t_i^s + \alpha_1 d_i]$ and the end region as $r_i^e = [t_i^e - \alpha_1 d_i, t_i^e + \alpha_1 d_i]$, where $d_i = t_i^e - t_i^s$ is the duration of E_i^*, and α_1 is hyper-parameter to control the width of the region. The centre region of E_i^* is defined as $r_i^c = [t_i^c - \alpha_2 d_i, t_i^c + \alpha_2 d_i]$, where $t_i^c = (t_i^s + t_i^e)/2$ is the centre timestamp of E_i^* and α_2 is the width-controller for centre regions. For the t-th snippet in V, its corresponding video timestamps are $r_t = [\tau_v(t-0.5), \tau_v(t+0.5)]$, where τ_v is the length of a snippet. We calculate the intersection-over-are (IoA) between r_t and r_i^s, r_i^c, or r_i^e for all snippets to generate labels $\boldsymbol{c}_s^*$, $\boldsymbol{c}_c^*$, and $\boldsymbol{c}_e^*$: if the IoA is larger than a threshold θ, then the label is 1 otherwise 0.

Therefore, with the predicted sequence $\boldsymbol{c}_x$ and the ground-truth sequence $\boldsymbol{c}_x^*$ ($x \in \{s, c, e\}$), we minimise the balanced logistic regression loss used in [15] for start, centre and end region prediction:

$$L_g = \frac{1}{T} \sum_{x \in \{s,c,e\}} \sum_{t=1}^{T} (\alpha_x^+ \boldsymbol{c}_{x,t}^* \log \boldsymbol{c}_{x,t} + \alpha_x^- (1 - \boldsymbol{c}_{x,t}^*) \log (1 - \boldsymbol{c}_{x,t})), \tag{9}$$

where $\alpha_x^+ = \frac{T}{\sum_t \boldsymbol{c}_{x,t}^*}$ and $\alpha_x^- = \frac{T}{T - \sum_t \boldsymbol{c}_{x,t}^*}$ are the balance weights for positive and negative samples.

Inference. For each query description S_i, we first generate its confidence scores $\boldsymbol{c}_s, \boldsymbol{c}_c, \boldsymbol{c}_e$ for all snippets in V. Then we enumerate all valid combinations of start and end indexes in the video, *i.e.* $[\iota_s, \iota_e]$ is valid if $\iota_e \geq \iota_s$. The confidence score of the candidate segment $[\iota_s, \iota_e]$ is $\boldsymbol{c}_{s,\iota_s} + \boldsymbol{c}_{e,\iota_e} + \boldsymbol{c}_{c,\iota_c}$, where ι_c is the centre index of the segment. The event segment with the largest confidence score is selected as the grounding prediction E_i.

3.5. Refining: Fine-grained Sentence Decoder

The coarse paragraph P obtained in the sketching phase is not designed to describe a specific event, which may fail to capture fine-grained event-specific details. Therefore, we further design a fine-grained sentence decoder $\mathcal{D}_2$ to refine sentences in P with the help of grounded video segments $\{E_i\}_{i=1}^n$ from $\mathcal{G}$.

We design a Dual-Path Cross Attention module (DPCA) for $\mathcal{D}_2$, which dynamically attends on both the coarse-grained sentence S_i and its aligned video segment E_i to generate a fine-grained event description. The awareness of specific video segment E_i encourages the model to be focused and generate more event-level details; while S_i serves as a reference for the refine process. Specifically, for S_i, we reuse its feature $\boldsymbol{H}_{s_i}$ from $\mathcal{D}_1$; for E_i, we reuse the $\boldsymbol{H}_v$ from $\mathcal{E}$ and crop it according to the boundary of E_i, denote as $\boldsymbol{H}_{e_i} \in \mathbb{R}^{T_{e_i} \times d_m}$, where T_{e_i} is the number of snippets inside E_i. $\mathcal{D}_2$ is a transformer-based decoder similar to $\mathcal{D}_1$, but is equipped with DPCA (see Figure 3) to incorporate both $\boldsymbol{H}_{e_i}$ and $\boldsymbol{H}_{s_i}$ during decoding. The computation in Eq. 4 is modified as follows:

$$\begin{aligned} \boldsymbol{H}_r^{l,c} = \boldsymbol{H}_r^{l,c} &+ \text{MHA}(\boldsymbol{H}_r^{l,s}, \boldsymbol{H}_{s_i}, \boldsymbol{H}_{s_i}) \\ &+ \text{MHA}(\boldsymbol{H}_r^{l,s}, \boldsymbol{H}_{e_i}, \boldsymbol{H}_{e_i}) \end{aligned} \tag{10}$$

where $\boldsymbol{H}_r^{l,*}$ denotes the hidden states of $\mathcal{D}_2$.

Training. Given a generated sentence S_i from $\mathcal{D}_1$ and its localised segment E_i, we use the ground-truth sentence in P^* whose segment has the highest intersection-over-union (IoU) score with E_i as the reference sentence in training. Denoting the selected reference sentence as S_i^* and its ground-truth video segment as E_i^*. The refinement decoder $\mathcal{D}_2$ is trained on triplets of (E_i, S_i, S_i^*). When there is only small overlap between E_i and E_i^*, the input video segment E_i can be mismatched with S_i^* which may harm the refinement. In this case, we shift the boundaries of E_i to increase its overlap with E_i^*. The shift is limited so that the new E_i has at least 0.5 IoU with the original E_i.

We first train $\mathcal{D}_2$ with the teacher forcing scheme as Eq. 6. In order to improve the refining performance, we propose to further fine-tune $\mathcal{D}_2$ with **refinement-enhanced rewards** in reinforcement learning (RL). The RL training aims to minimise the negative reward $L_r(S_i^r) = -\mathbb{E}_{S_i^r \sim \pi}[R(S_i^r)]$, where S_i^r is a refined sentence generated via policy π, and $R(\cdot)$ is a non-differential reward function, *i.e.* the METEOR score [1] that measures similarity between S_i^r and $\hat{S}_i^*$. The policy π randomly samples sentence according to polynomial word distribution predicted from $\mathcal{D}_2$. The model can be optimised through policy gradient [26] as follows:

$$\nabla L_r(S_i^r) = -(R(S_i^r) - b) \nabla \log \pi(S_i^r), \tag{11}$$

where b is a baseline to stabilise training. In self-critical RL [23], $b = R(\bar{S}_i^r)$ is the reward of a greedy-decoded sentence $\bar{S}_i^r$. Thus, the model is encouraged to generate sentences that have higher rewards than $\bar{S}_i^r$. The term $R(S_i^r) - b$ is referred as the advantage function in the RL literature.

We propose a new advantage function to encourage the sentence refinement, which explicitly considers two goals. Firstly, the refined sentence S_i^r should have a better quality than the coarse-grained sentence S_i from $\mathcal{D}_1$ in sketching, besides being better than the greedy-decoded sentence

Method	B@1	B@4	C	M
DCE [11]	10.81	0.71	12.43	5.69
TDA-CG [33]*	10.75	1.31	7.99	5.86
DVC [13]	12.22	0.73	12.61	6.93
MFT [36]	13.31	1.24	21.00	7.08
SDVC [18]	**17.92**	0.93	**30.68**	8.82
SG	13.68	1.56	21.54	7.79
SG w/ refine caption	13.83	1.63	21.87	8.85
SGR	14.05	**1.67**	22.12	**9.07**

Table 1. Dense video captioning results (using C3D feature) of our model and state-of-the-art methods on BLEU@N (B@N), CIDEr (C) and METEOR (M) on ActivityNet Captioning validation set. $*$ indicates the results re-calculated by new evaluation tool.

$\bar{S}_i^r$ of itself. Hence, the refine-aware advantage function is computed as:

$$\mathrm{A}_{s_i} = R(S_i^r) - R(\bar{S}_i^r) + R(S_i^r) - R(S_i) \tag{12}$$

Secondly, all refined sentences should convey a coherent story when concatenated together. Therefore, we also leverage a paragraph-level advantage function:

$$\mathrm{A}_p = R(P^r) - R(\bar{P}^r) + R(P^r) - R(P) \tag{13}$$

P^r and $\bar{P}^r$ are the concatenation of S_i^r and $\bar{S}_i^r$, respectively, and P is from $\mathcal{D}_1$. Each input (E_i, S_i) is optimised by advantage function $A_{s_i} + A_p$ in RL training.

Inference. We use beam search to generate refined sentence S_i^r given S_i in the paragraph and its aligned video segment E_i. Since the refined sentence is more event-specific, we further input the refined sentences into $\mathcal{G}$ again to adjust event boundaries as shown in Algorithm 1.

4. Experiment

In this section, we evaluate the proposed method on the benchmark ActivityNet Captioning dataset [11]. It contains 19,994 YouTube videos separated into three subsets with 10,024, 4,926, and 5,044 videos for training, validation, and testing, respectively. The videos have an average length of 120 seconds and have 3.65 temporally localised events with corresponding descriptions on average. The average length of the descriptions is 13.48 words. We consider two types of video feature extractor during our experiments: 1) a C3D [28] network, which is mostly-used in previous DVC methods; 2) a TSN-like network [37], where a ResNet200 [8] is used to extract appearance feature and a BN-Inception [9] is used to extract optical flow feature.

4.1. Evaluation

For evaluation, we use the evaluation tool provided by the 2018 ActivityNet Captions Challenge, which measures the capability to localise and describe events. Specifically,

Method	ActivityNet		YouCook2
	METEOR	SODA	METEOR
Masked-Transformer [43]*	4.98	4.02	3.18
SG	8.27	4.83	3.67
SG w/ refine caption	9.13	4.99	3.96
SGR	**9.37**	**5.29**	**4.35**

Table 2. Dense video captioning results (using TSN feature) on two datasets in terms of the standard evaluation metric (METEOR) and the story-oriented evaluation metric [6] (SODA).

we measure the METEOR [1], CIDEr [30] and BLEU [22] scores of the dense video captions. Following the common practice, we use METEOR as the primary metric for comparison. The scores of the metrics are summarised via their averages based on IoU thresholds of 0.3, 0.5, 0.7, and 0.9 given event captions and the corresponding event segments.

4.2. Implementation Details

Following the original Transformer [29] implementation, the number of layers in $\mathcal{G}$, $\mathcal{D}_1$ and $\mathcal{D}_2$ are all set to 6. The hidden size d_m is 512, and the number of attention heads is 8. The dimension of the 1-D convolutional layers in $\mathcal{G}$ is set to 256. The α_1 and α_2 used for controlling the region width when training the grounding stage are set to 0.1 and 0.3, respectively. The IoA threshold θ is set to 0.5. During the first training step, the balancing coefficient λ is set to 1, and the model is trained using AdamW [16] for 50 epochs with a batch size of 16 and learning rate of 1e-4. The label smoothing factor is set to 0.1 and the scheduled sampling probability is set to 0 at the start of training and linearly increased to 0.3 in the end. At the second training step, the model is first trained with teacher forcing scheme for 20 epochs, and then switch to reinforcement learning for another 50 epochs. The AdamW optimiser is also adopted at this step and the batch size is kept as 16, while the learning rate is reduced to 2e-5.

4.3. Comparisons with state of the arts

We first compare the proposed method with several state-of-the-art DVC methods. Specifically, for the proposed model, we report the performance of: 1) Sketch and Ground (SG), *i.e.*, without the refinement stages; 2) SG with the caption refinement but without the second grounding stage; 3) SG with both caption and segment refinement, *i.e.*, the full SGR model. C3D video feature extractor is used for fair comparisons. As shown in Table 1, the SG model achieves the second-best METEOR score (7.79) among previous methods, showing the competitiveness of our basic top-down DVC framework. Moreover, by refining the event captions, the performance of our model improves significantly to 8.85. After further adjusting the event segment boundaries according to the refined event captions, the pro-

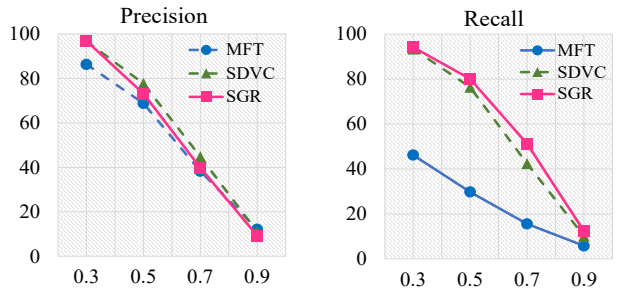

Figure 4. Event localisation performances at four IoU threshold (using C3D feature) on the ActivityNet Captioning validation set. The results are reported in percentage.

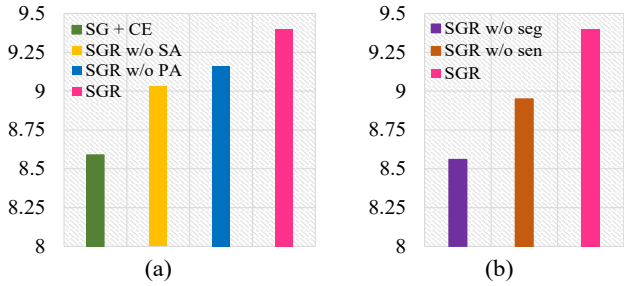

Figure 5. Ablation studies (using TSN feature) on advantage functions (a) and DPCA branches (b) in terms of METEOR score. "SG + CE" stands for the model refined by cross entropy loss. "SA" and "PA" stand for the sentence-level and paragraph-level advantages, respectively. "sen" and "seg" stand for the sentence and segment branches, respectively.

posed SGR model outperforms the previous best model by a clear margin on METEOR score[1]. These results quantitatively verify the effectiveness of the proposed method.

Moreover, we follow [43] and evaluate the performance of our method with a more advanced TSN feature extractor on the ActivityNet Captioning dataset and an additional YouCook2 [42] dataset. YouCook2 is relatively smaller compared to ActivityNet Captioning, with 2,000 videos focusing on 89 cooking activities. We further compare models under a newly proposed evaluation metric (SODA [6]). SODA is a story-oriented dense captioning metric that tries to find temporally optimal matching between generated and reference captions to capture the story structure of the dense video captions. Since SODA is sensitive to the number of dense captions per video, for fair comparison, we obtain the baseline results by setting the total number of captions equal to the number of ground-truth event captions, *i.e.*, roughly equal to the number of captions generated by our proposed SGR. The results are shown in Table 2. From the table, our proposed models consistently outperform the baseline model [43] in all metrics. Besides, SGR introduces clear gains over SG in all settings, which is identical to the observation we obtained from Table 1. These results show the superiority of our top-down DVC framework.

We also evaluate the event localisation quality of our model in terms of recall and precision at four IoU thresholds. As shown in Figure 4, the proposed SGR model achieves a better recall rate than the previous best method SDVC and also obtain comparable performance with SDVC in terms of precision rate. This shows that the top-down SGR can localise the events in a video accurately without proposing hundreds or even thousands of event proposals.

4.4. Performance on Video Sentence Localisation

We further evaluate the video sentence localisation performance of our grounding module using *ground-truth* event captions. C3D feature is used to align with previous methods [39, 17, 40]. The model performance is measured by localisation accuracy, where an event caption is considered to be correctly localised only when the IoU score between one of its top-K predicted segments and the ground-truth segment is higher than a threshold Θ. Denote this measure as Top$K$@Θ. As mentioned in Section 3.4, compared with the standard setting of video sentence localisation adopted in the compared baseline methods, *i.e.*, only the caption of the target event is available to the model, our grounding module further leverages the context information from the whole video paragraph when localising each individual event caption, which is beneficial since it enables our grounding module to be aware of the relative order of the events in the video. As shown in Table 3, our grounding module indeed yields a significant performance improvement over the baseline models, *e.g.*, our model improves the Top1@0.5 and Top1@0.7 performance over the state-of-the-art model (DRN) by 15.1 and 13.8 percent, respectively. This suggests that the grounding stage plays a reliable role in our dense video captioning framework.

Method	Top1@0.5	Top1@0.7	Top5@0.5	Top5@0.7
SCDM [39]	36.75	19.86	64.99	41.53
SQAN [17]	41.51	23.07	-	-
DRN [40]	42.49	22.25	71.85	45.96
Ours	**57.63**	**36.02**	**75.71**	**60.25**

Table 3. Video sentence localisation results (using C3D feature) on ActivityNet Captioning validation set. Reported in percentage.

4.5. Ablation Studies

In the ablation studies, we first evaluate the effectiveness of the advantage functions introduced in Eq.(12) and Eq.(13), respectively. When training the ablation model, one of the advantage functions is removed during the re-

[1]We tried to further repeat the refinement stage multiple times but the improvements are minor. We guess this is because the grounding stage is good enough, which can not further improve the captioning ability.

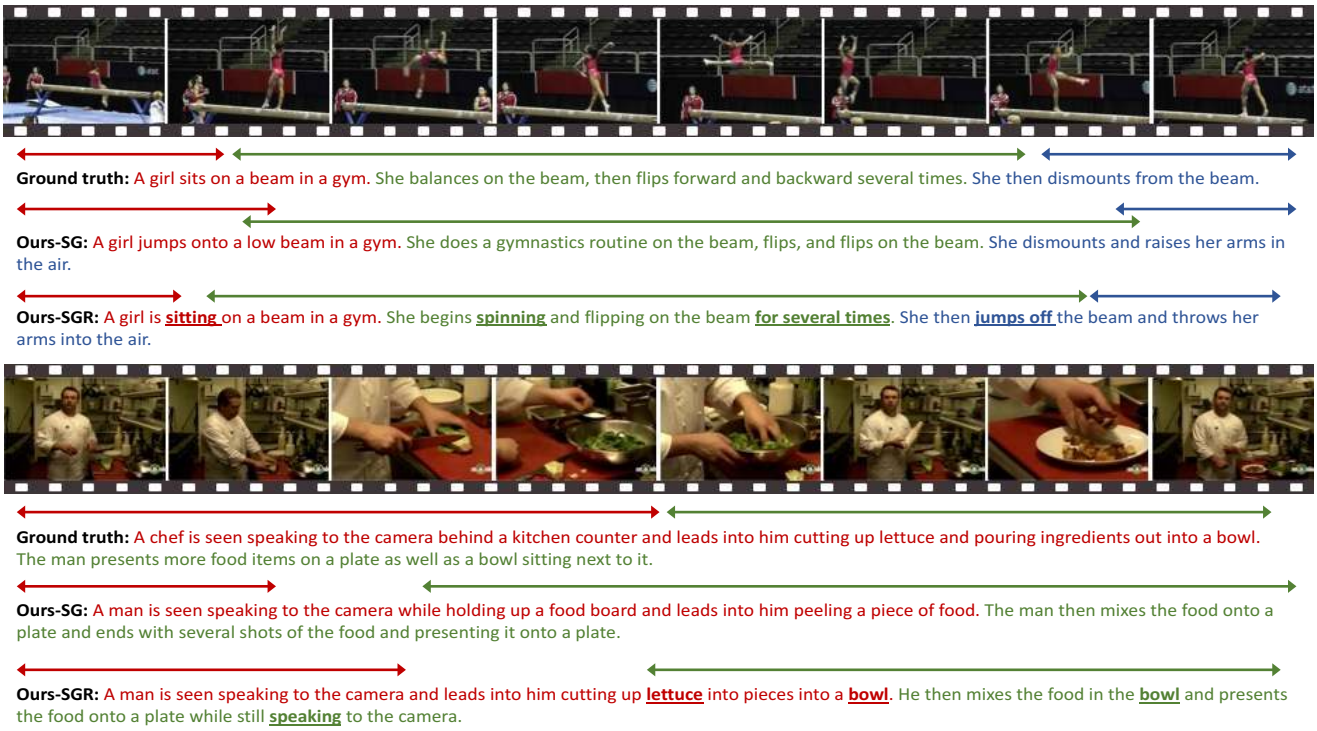

Figure 6. Qualitative results on ActivityNet Captioning. The coloured bars represent different events. Coloured text indicates the relevant description of the events. The lined words show the detailed information introduced by the caption refinement stage. Best viewed in colour.

inforcement learning phrase. The results are shown in Figure 5(a). Based on the experiments, removing any of the advantage functions causes 0.2-0.3 performance degradation, while removing both of them (SG+CE) further drops the performance by more than 0.4. This demonstrates the effectiveness of the RL-training, and also shows the importance of using hierarchical reward signals, which may alleviate the credit assignment problem in reinforcement learning.

We also evaluate the importance of the proposed DPCA module by removing one of its cross attention branches. As shown in Figure 5(b), the final METEOR score drops significantly when removing any of the branches, especially the segment branch. This result suggests that both the coarse sentences and coarse segments are useful knowledge to the refinement process.

4.6. Qualitative Results

Qualitative results of the proposed model including SG and SGR are presented in Figure 6. From the results, the SG model describes the video with coherent event descriptions and also effectively localise the event segments. Moreover, the SGR model introduces more fine-grained information into the captions than the SG model, and describes the events with more details. As a result, the localisation accuracy of the event segments is improved accordingly.

5. Conclusion

In this paper, we propose a top-down dense video captioning framework termed "Sketch, Ground, and Refine" (SGR), which first generates a video-level story and then grounds the story to video segments for further refinement. In this way, we avoid the ill-defined event proposal generation process and directly discover a sequence of story-oriented events in the video, thus improving the coherency and accuracy of the generated event captions. To facilitate the event captions to contain more event-specific details, we adopt a refinement stage to leverage the event-level information and introduce more fine-grained details into the event captions. Based on the refined captions, the event segments can further be adjusted to be more accurate. In the experiments, the proposed SGR model outperforms the previous state-of-the-art methods on both traditional and story-oriented dense caption evaluations.

6. Acknowledgement

This work is supported by Alibaba Group through Alibaba Research Intern Program. Qi Wu is supproted by ARC DECRA DE190100539.

References

[1] Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, 2005. 5, 6

[2] Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. In *Advances in Neural Information Processing Systems*, pages 1171–1179, 2015. 4

[3] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 961–970, 2015. 1

[4] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 1

[5] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6202–6211, 2019. 1

[6] Soichiro Fujita, Tsutomu Hirao, Hidetaka Kamigaito, Manabu Okumura, and Masaaki Nagata. Soda: Story oriented dense video captioning evaluation flamework. In *Proceedings of the European Conference on Computer Vision*, August 2020. 6, 7

[7] Lianli Gao, Kaixuan Fan, Jingkuan Song, Xianglong Liu, Xing Xu, and Heng Tao Shen. Deliberate attention networks for image captioning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8320–8327, 2019. 3

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 6

[9] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pages 448–456, 2015. 6

[10] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 1

[11] 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. 1, 2, 3, 6

[12] Jie Lei, Liwei Wang, Yelong Shen, Dong Yu, Tamara L Berg, and Mohit Bansal. Mart: Memory-augmented recurrent transformer for coherent video paragraph captioning. In *Proceedings of the 58th annual meeting on Association for Computational Linguistics*, 2020. 2

[13] Yehao Li, Ting Yao, Yingwei Pan, Hongyang Chao, and Tao Mei. Jointly localizing and describing events for dense video captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7492–7500, 2018. 1, 2, 3, 6

[14] Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen. Bmn: Boundary-matching network for temporal action proposal generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3889–3898, 2019. 5

[15] Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang. Bsn: Boundary sensitive network for temporal action proposal generation. In *Proceedings of the European Conference on Computer Vision*, pages 3–19, 2018. 1, 5

[16] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2018. 6

[17] Jonghwan Mun, Minsu Cho, and Bohyung Han. Local-global video-text interactions for temporal grounding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10810–10819, 2020. 7

[18] Jonghwan Mun, Linjie Yang, Zhou Ren, Ning Xu, and Bohyung Han. Streamlined dense video captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6588–6597, 2019. 1, 2, 3, 6

[19] Pingbo Pan, Zhongwen Xu, Yi Yang, Fei Wu, and Yueting Zhuang. Hierarchical recurrent neural encoder for video representation with application to captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1029–1038, 2016. 2

[20] Yingwei Pan, Tao Mei, Ting Yao, Houqiang Li, and Yong Rui. Jointly modeling embedding and translation to bridge video and language. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4594–4602, 2016. 2

[21] Yingwei Pan, Ting Yao, Houqiang Li, and Tao Mei. Video captioning with transferred semantic attributes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6504–6512, 2017. 2

[22] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on Association for Computational Linguistics*, pages 311–318. Association for Computational Linguistics, 2002. 6

[23] Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. Self-critical sequence training for image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7008–7024, 2017. 5

[24] Dian Shao, Yu Xiong, Yue Zhao, Qingqiu Huang, Yu Qiao, and Dahua Lin. Find and focus: Retrieve and localize video events with natural language queries. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 200–216, 2018. 3

[25] Zheng Shou, Dongang Wang, and Shih-Fu Chang. Temporal action localization in untrimmed videos via multi-stage cnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1049–1058, 2016. 1

[26] Richard S Sutton, David A McAllester, Satinder P Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems*, pages 1057–1063, 2000. 5

[27] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2818–2826, 2016. 4

[28] 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. 6

[29] 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. 2, 4, 6

[30] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4566–4575, 2015. 6

[31] Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. Sequence to sequence-video to text. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4534–4542, 2015. 2

[32] Bairui Wang, Lin Ma, Wei Zhang, and Wei Liu. Reconstruction network for video captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7622–7631, 2018. 2

[33] Jingwen Wang, Wenhao Jiang, Lin Ma, Wei Liu, and Yong Xu. Bidirectional attentive fusion with context gating for dense video captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7190–7198, 2018. 1, 2, 6

[34] Junbo Wang, Wei Wang, Yan Huang, Liang Wang, and Tieniu Tan. M3: Multimodal memory modelling for video captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7512–7520, 2018. 2

[35] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *Proceedings of the European Conference on Computer Vision*, pages 20–36. Springer, 2016. 1

[36] Yilei Xiong, Bo Dai, and Dahua Lin. Move forward and tell: A progressive generator of video descriptions. In *Proceedings of the European Conference on Computer Vision*, pages 468–483, 2018. 2, 6

[37] Yuanjun Xiong, Limin Wang, Zhe Wang, Bowen Zhang, Hang Song, Wei Li, Dahua Lin, Yu Qiao, Luc Van Gool, and Xiaoou Tang. Cuhk & ethz & siat submission to activitynet challenge 2016. *arXiv preprint arXiv:1608.00797*, 2016. 6

[38] Haonan Yu, Jiang Wang, Zhiheng Huang, Yi Yang, and Wei Xu. Video paragraph captioning using hierarchical recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4584–4593, 2016. 2

[39] 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. 7

[40] Runhao Zeng, Haoming Xu, Wenbing Huang, Peihao Chen, Mingkui Tan, and Chuang Gan. Dense regression network for video grounding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10287–10296, 2020. 7

[41] 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

[42] Luowei Zhou, Chenliang Xu, and Jason J Corso. Towards automatic learning of procedures from web instructional videos. In *AAAI Conference on Artificial Intelligence*, pages 7590–7598, 2018. 2, 7

[43] Luowei Zhou, Yingbo Zhou, Jason J Corso, Richard Socher, and Caiming Xiong. End-to-end dense video captioning with masked transformer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8739–8748, 2018. 1, 2, 6, 7