

# Jointly Localizing and Describing Events for Dense Video Captioning\*

Yehao Li <sup>†¶</sup>, Ting Yao <sup>‡</sup>, Yingwei Pan <sup>§</sup>, Hongyang Chao <sup>†¶</sup>, and Tao Mei <sup>‡</sup>

<sup>†</sup> School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China

<sup>‡</sup> Microsoft Research, Beijing, China

<sup>§</sup> University of Science and Technology of China, Hefei, China

<sup>¶</sup> Key Laboratory of Machine Intelligence and Advanced Computing (Sun Yat-sen University), Ministry of Education

{yehaoli.sysu, panyw.ustc}@gmail.com, {tiyao, tmei}@microsoft.com, isschhy@mail.sysu.edu.cn

## Abstract

Automatically describing a video with natural language is regarded as a fundamental challenge in computer vision. The problem nevertheless is not trivial especially when a video contains multiple events to be worthy of mention, which often happens in real videos. A valid question is how to temporally localize and then describe events, which is known as “dense video captioning.” In this paper, we present a novel framework for dense video captioning that unifies the localization of temporal event proposals and sentence generation of each proposal, by jointly training them in an end-to-end manner. To combine these two worlds, we integrate a new design, namely descriptiveness regression, into a single shot detection structure to infer the descriptive complexity of each detected proposal via sentence generation. This in turn adjusts the temporal locations of each event proposal. Our model differs from existing dense video captioning methods since we propose a joint and global optimization of detection and captioning, and the framework uniquely capitalizes on an attribute-augmented video captioning architecture. Extensive experiments are conducted on ActivityNet Captions dataset and our framework shows clear improvements when compared to the state-of-the-art techniques. More remarkably, we obtain a new record: ME-TEOR of 12.96% on ActivityNet Captions official test set.

## 1. Introduction

The recent advances in 2D and 3D Convolutional Neural Networks (CNNs) have successfully pushed the limits and improved the state-of-the-art of video understanding. For instance, the first rank performance achieves 8.8% in terms of top-1 error in untrimmed video classification task of ActivityNet Challenge 2017 [9]. As such, it has become possible to recognize a video with a pre-defined set of labels

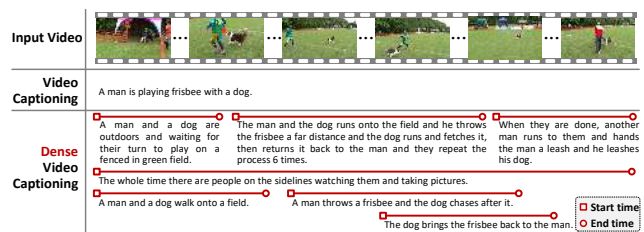


Figure 1. Examples of video captioning and dense video captioning (upper row: input video; middle row: the sentence generated by video captioning method; bottom row: temporally localized sentences generated by dense video captioning approach.)

or categories. In a further step to describe a video with a complete and nature sentence, video captioning [19, 32, 34] has expanded the understanding from individual labels to a sequence of words to express richer semantics and relationships in the video. Nevertheless, considering that videos in real life are usually long and contain multiple events, the conventional video captioning methods generating only one caption for a video in general will fail to recapitulate all the events in the video. Take the video in Figure 1 as an example, the output sentence generated by a popular video captioning method [32] is unable to describe the procedure of “playing frisbee with a dog” in detail. As a result, the task of dense video captioning is introduced recently in [16] and the ultimate goal is to generate a sentence for each event occurring in the video.

The difficulty of dense video captioning originates from two aspects: 1) how to accurately localize each event in time? 2) how to design a powerful sentence generation model? In the literature, there have been several techniques, including temporal action proposal [2, 3, 6, 17] and image/video captioning [32, 34, 37], being proposed for each individual aspect. However, simply solving the problem of dense video captioning in a two-stage way, i.e., first temporal event proposal and then sentence generation, may destroy the interaction between localizing and describing events, resulting in a sub-optimal solution.

This paper proposes a novel deep architecture to unify

\*This work was performed at Microsoft Research Asia.

the accurate localization of temporal events with the descriptive principle of sentence generation for dense video captioning. Technically, we devise a new descriptiveness regression component and integrate it into a single shot detection framework as a bridge, on one hand to measure the complexity of each event being described in sentence generation, and on the other, to adjust the event proposal. More specifically, the descriptiveness regression guides the learning of temporal event proposal together with event/background classification and temporal coordinate regression. In between, event/background classification is to predict event proposals and temporal coordinate regression is to refine the temporal boundaries of each proposal. Furthermore, the inference of descriptiveness regression is employed as an “attention” to weight video clips in each proposal locally. The proposal-level representation is then averaged over all the clip-level representations in the proposal weighted by the holistic attention score of each clip and finally fed into an attribute-augmented captioning architecture for sentence generation. As such, the task of dense video captioning could be jointly learnt and globally optimized in an end-to-end manner.

The main contribution of this work is the proposal of a new architecture to unify the temporal localization of event proposals and sentence generation for dense video captioning. The solution also leads to the elegant views of what kind of interaction should be built across the two sub problems and how to model and integrate the interaction in a deep learning framework, which are problems not yet fully understood in the literature.

## 2. Related Work

**Temporal Action Proposal.** [5] is one of the early works that detects temporal segments containing the action of interest in a sliding windows fashion. Next, a few subsequent works [2, 3, 6, 8, 27] tackle temporal action proposal by leveraging action classifiers on a smaller number of temporal windows. In particular, Sparse-prop [3] utilizes dictionary learning to encode representations of trimmed action instances and then retrieves the most representative segments from testing videos, which are treated as the class-independent proposals. S-CNN [27] trains a 3D CNNs to classify a video segment as background or being-action and employs varied length temporal windows for multi-scale action proposal generation. Later in [6], DAPs utilizes Long Short-Term Memory (LSTM) to encode video streams and enables multi-scale proposal generation inside the streams with a single pass through the video, obviating the need for deploying sliding windows on multiple scales. Furthermore, Buch *et al.* develop SST based on DAPs by constructing no overlapping sliding windows over the input video and encoding each window sequentially with a Gated Recurrent Unit (GRU) in [2]. Most recently, Gao *et al.* [8]

design temporal coordinate regression for temporal action proposal generation.

**Video Captioning.** The research in this direction has proceeded along two different dimensions: template-based language methods [11, 15, 26] and sequence learning approaches (e.g., RNNs) [16, 19, 32, 33, 34, 38]. Template-based language methods directly generate the sentence with detected keywords in predefined language template. Sequence learning approaches utilize CNN plus RNN architecture to generate novel sentences with more flexible syntactical structures. In [33], Venugopalan *et al.* present a LSTM-based model to generate video descriptions with the mean pooling representation over all frames. The framework is then extended by inputting both frames and optical flow images into an encoder-decoder LSTM in [32]. Compared to mean pooling, Yao *et al.* propose to utilize the temporal attention mechanism to exploit temporal structure for video captioning [34]. Later in [38], a hierarchical RNN is devised to further capture the inter-sentence dependency, targeting for describing a long video with a paragraph consisting of multiple sentences. Different from the video paragraph captioning with non-overlapping and annotated temporal intervals, a more challenge task, named as dense video captioning, is recently introduced in [16] which involves both detecting and describing multiple events in a video. A two-stage dense-captioning system is thus designed by leveraging DAPs [6] to localize temporal event proposals and a LSTM-based sequence learning module to describe each event proposal. Most recently, [35] additionally incorporates KNN-based retrieval module into LSTM-based sequence learning module to boost video captioning.

**Summary.** Our work aims to detect and describe events in video, i.e., dense video captioning. Different from the aforementioned method [16], our approach contributes by studying not only detecting the events with the simple objective of binary classification (i.e., event or background) and modeling sentence generation with LSTM, but also enhancing the temporal event proposal by utilizing both temporal boundary regression to correct start and end time of event and descriptiveness regression to infer whether the event can be well described from language perspective. Moreover, sentence generation module is further boosted by leveraging semantic attributes and reinforcement learning to optimize LSTM with non-differentiable metrics.

## 3. Dense Video Captioning

The basic idea of this work is to automatic describe multiple events in videos by temporally localizing event proposals and generating language sentence for each event proposal. The temporal event proposal (TEP) is performed by encapsulating the event classification to recognize video segments of events from backgrounds, proposal generation to temporally localize the event, and descriptiveness inference

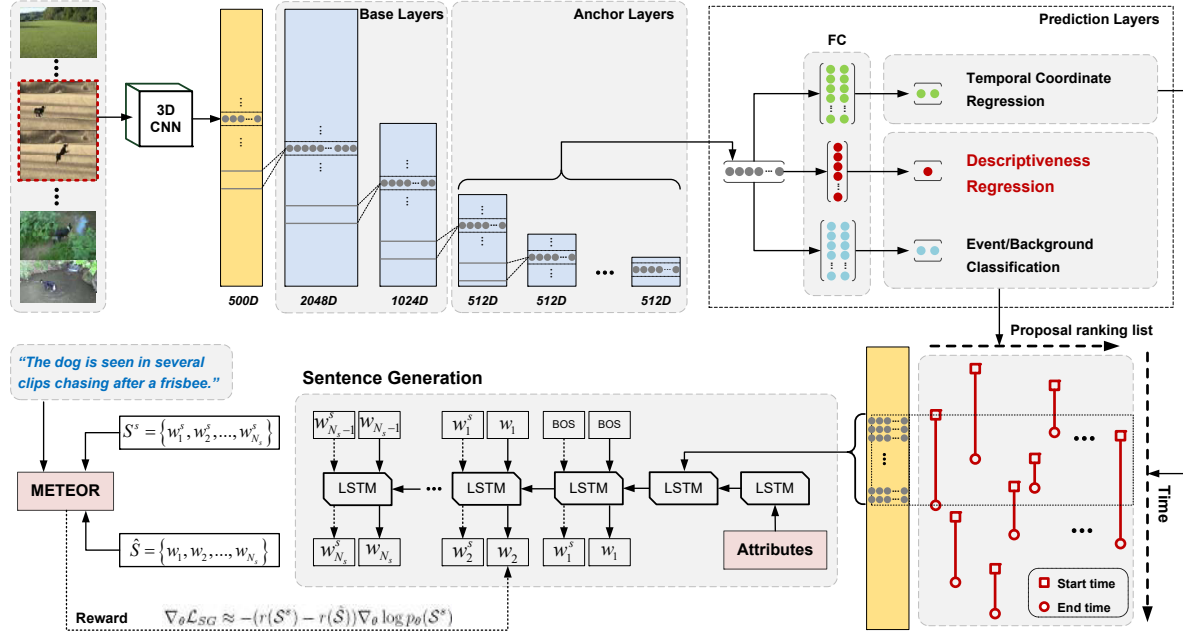


Figure 2. An overview of our Dense Video Captioning framework mainly including Temporal Event Proposal (TEP) and Sentence Generation (SG) (better viewed in color). The input video is first encoded into a series of clip-level features via a 3-D CNNs, which are fed into TEP module to produce candidate proposals. The TEP module is employed by integrating the event/background classification to predict event proposal, temporal coordinate regression to refine the temporal boundaries of each proposal, and descriptiveness regression to infer the descriptive complexity of each event, into a single shot detection architecture. After ranking the candidate proposals with regard to both eventness and descriptiveness scores, the top proposals are in turn injected into SG module for sentence generation. The SG module leverages both attributes and reinforcement learning based optimization to enhance captioning. The whole system can be trained through the joint and global optimization of detection and captioning in an end-to-end manner.

procedure to infer the descriptive complexity of this event, in one single network. Such design enables straightforward temporal event proposal in a single shot manner to ease the training consumption. The sentence generation module (SG) leverages an attribute-augmented LSTM-based model for generating descriptions. Moreover, the policy-gradient based reinforcement learning is adopted to optimize LSTM with evaluation metric based reward, harmonizing the module with respect to the testing inference. Please note that the descriptiveness inference procedure in TEP module is not only leveraged to additionally refine the localized event proposal from language perspective through descriptiveness regression, but also integrated into SG module to consider the descriptiveness scores as one kind of temporal attention over clips for weighted fusing them as the input proposal-level representation of LSTM. As such, our system including both TEP and SG modules can be jointly trained through the global optimization of detection and captioning in an end-to-end manner. An overview of our dense video captioning system is illustrated in Figure 2.

### 3.1. Problem Formulation

Suppose we have a video  $\mathcal{V} = \{v_t\}_{t=1}^{T_v}$  with  $T_v$  frames/clips and  $v_t$  denotes the  $t$ -th frame/clip in temporal order. The ultimate target of our dense video captioning

system is to generate a set of temporal localized descriptions  $\Phi_v = \{\phi_i = (t_{start}^i, t_{end}^i, \mathcal{S}_i)\}_{i=1}^{M_v}$  for the input video  $\mathcal{V}$ , where  $M_v$  is the number of sentences,  $t_{start}^i$  and  $t_{end}^i$  represent the starting time and ending time for each sentence  $\mathcal{S}_i$ , and  $\mathcal{S}_i = \{w_1, w_2, \dots, w_{N_s}\}$  consists of  $N_s$  words.

Hence the TEP module in our system is firstly utilized to produce a set of candidate proposals for the input video  $\mathcal{V}$ :

$$\Phi_p = \{\phi_p^i = (t_{start}^i, t_{end}^i, p_{event}^i, p_{des}^i)\}_{i=1}^{N_p}, \quad (1)$$

where  $p_{event}^i$  is the probability of recognizing the candidate as an event (i.e., eventness score),  $p_{des}^i$  denotes the descriptiveness score measuring how well the candidate can be described from language perspective and  $N_p$  is the total number of candidate proposals. By consolidating the idea of selecting proposals from both vision and language perspectives, all the candidates are ranked according to the fused score  $p_{conf}^i = p_{event}^i + \lambda_0 p_{des}^i$  and only the candidates with a  $p_{conf}^i$  higher than a threshold are injected into SG module for captioning, denoted as  $\Phi_{\hat{p}}$ .

Inspired by the successes of sequence learning models in machine translation [28] and attributes utilized in image/video captioning [7, 20, 36], we formulate our SG module as an attribute-argument LSTM-based model which encodes the input event proposal representation (**F**) and its detected attributes/categories (**A**) into a fixed dimensional vector and then decodes it to the output target sentence. As

such, the sentence generation problem we exploit here can be formulated by minimizing the negative log probability of the correct textual sentence ( $-\log \Pr(\mathcal{S}|\mathbf{F}, \mathbf{A})$ ). The negative log probability is typically measured with cross entropy loss, resulting in the discrepancy of evaluation between training and inference. Hence, to further boost our SG module by amending such discrepancy, we take the inspiration from the reinforcement learning [23] leveraged in sequence learning and directly optimize LSTM by minimizing the following expected sentence-level reward loss as

$$\mathcal{L}_{SG} = -\mathbb{E}_{\mathcal{S} \sim p_\theta}[r(\mathcal{S})], \quad (2)$$

where  $\theta$  denotes the parameters of LSTM that schedule a policy  $p_\theta$  for generating sentence.  $r(\mathcal{S})$  is the reward measured by comparing the generated sentence  $\mathcal{S}$  to ground-truth sentences over non-differentiable evaluation metric.

### 3.2. Temporal Event Proposal

Existing solutions for temporal action/event proposal mainly focus on detecting events with binary classifier (i.e., event or background) in a sliding windows fashion. However, the temporal sliding windows are typically too dense and even designed with multiple scales, resulting in the heavy computation cost. Inspired from spatial boundary regression in object localization [24] which simultaneously predicts objectness score and object bound, we integrate the event classification with temporal coordinate regression for correcting event’s temporal bound, pursuing both low-cost and high-quality event proposals. Moreover, for the dense video captioning task, the event identification undoubtedly plays the major role in temporal event proposal, while the proposals containing rich describable objects or scenes are also preferred by human beings in description generation. As such, a novel descriptiveness regression is especially devised to infer the descriptive complexity of each proposal, and further refine the event proposal from language perspective. Similar to single shot object detection in [18], all the three components (i.e., event classification, temporal coordinate regression and descriptiveness regression) are elegantly integrated into one feed-forward CNNs, aiming to simultaneously produce a fixed-size set of proposals, scores for the presence of event in the proposals (i.e., eventness scores), and descriptiveness scores of proposals.

**Input.** Technically, given input video  $\mathcal{V} = \{v_t\}_{t=1}^{T_v}$ , a 3-D CNN is utilized to encode the frame sequence into a series of clip-level features  $\{\mathbf{f}_t = F(v_t : v_{t+\delta})\}_{t=1}^{T_f}$  where  $\delta$  is the temporal resolution of each feature  $\mathbf{f}_t$ . In our experiments, C3D [30] is adopted as 3-D CNNs encoder  $F$  with  $\delta = 16$  frames and the temporal interval for encoding is set as 8 frames, resulting in the initial output feature map with the size of  $T_f \times D_0$ . Note that  $D_0$  is the dimension of clip-level feature and  $T_f = T_v/8$  discretizes the video frames.

**Network Architecture.** The initial feature map of size  $T_f \times D_0$  is fed into 1-D CNNs architecture in TEP mod-

ule, which consists of convolutional layers in three groups: base layers, anchor layers, and prediction layers, as shown in Figure 2. In particular, two base layers ( $conv_1$  and  $conv_2$ ) are firstly devised to reduce the temporal dimension of feature map and increase the size of temporal receptive fields, producing the output feature map of size  $T_f/2 \times 1024$ . Then, we stack nine anchor layers ( $conv_3$  to  $conv_{11}$ ) on the top of base layer  $conv_2$ , each of which is designed with the same configuration (kernel size: 3, stride size: 2, and filter number: 512). These anchor layers decrease in temporal dimension of feature map progressively, enabling the temporal event proposals at multiple temporal scales.

For each anchor layer, its output feature map is injected into prediction layer to produce a fixed set of predictions in one shot manner. Concretely, given an output feature map  $f_j$  with the size of  $T_{f_j} \times D_j$ , the basic element (anchor) for predicting parameters of a candidate proposal is a  $1 \times D_j$  feature map cell that produces a prediction score vector  $p_{pred} = (p_{cls}, \Delta c, \Delta w, p_{des})$  via fully connected layers.  $p_{cls} = [p_{event}, p_{bk}]$  denotes the two dimensional classification scores for event/background and  $p_{des}$  is the descriptiveness score to infer the confidence of this proposal to be well described.  $\Delta c$  and  $\Delta w$  are two temporal offsets relative to the default center location  $\mu_c$  and width  $\mu_w$  of this anchor, which are leveraged to adjust its temporal coordinates as

$$\begin{aligned} \varphi_c &= \mu_c + \alpha_1 \mu_w \Delta c, & \varphi_w &= \mu_w \exp(\alpha_2 \Delta w), \\ t_{start} &= \varphi_c - \frac{1}{2} \varphi_w, & t_{end} &= \varphi_c + \frac{1}{2} \varphi_w, \end{aligned} \quad (3)$$

where  $\varphi_c$  and  $\varphi_w$  are refined center location and width of the anchor.  $\alpha_1$  and  $\alpha_2$  are utilized to control the impact of temporal offsets, both of which are set as 0.1.  $t_{start}$  and  $t_{end}$  represent the adjusted starting and ending time of the anchor. In addition, derived from the anchor boxes in [18, 24], we associate a set of default temporal boundaries with each feature map cell. The different temporal scale ratios for these default temporal boundaries are denoted as  $R_s = \{r_d\}_{d=1}^{D_s} = \{1, 1.25, 1.5\}$ . For each temporal scale ratio  $r_d$ , we can thus achieve one default center location ( $\mu_{c_d} = \frac{t+0.5}{T_{f_j}}$ ) and width ( $\mu_{w_d} = \frac{r_d}{T_{f_j}}$ ) of  $t$ -th feature map cell, resulting in a total of  $T_{f_j} D_s$  anchors. Accordingly, for the feature map  $f_j$ , the set of all the produced proposals is defined as  $\Phi_{f_j} = \{\phi_{f_j} = (t_{start}^u, t_{end}^u, p_{cls}^u, p_{des}^u)\}_{u=1}^{N_U}$ , where  $N_U = T_{f_j} D_s$ . By accumulating all the produced proposals of the output feature maps of nine anchor layers, the final predicted proposal set is  $\Phi_p = \{\Phi_{f_j}\} = \{\phi_p^i\}_{i=1}^{N_p}$ .

**Training.** During training, a positive/negative label should be firstly assigned to each predicted proposal conditioned on the ground truth proposal set  $\Phi_v$ . Specifically, for each  $\phi_p^i \in \Phi_p$ , we measure its temporal Intersection over Union (tIoU) with each ground truth proposal and obtain the highest tIoU. If the highest tIoU is larger than 0.7,  $\phi_p^i$  is treated as positive sample with regard to the corresponding ground truth proposal  $\phi_g$ , otherwise  $\phi_p^i$  is a negative sample.

The training objective in our TEP module is formulated as a multi-task loss by integrating the event/background classification loss ( $\mathcal{L}_{event}$ ) for distinguishing events from backgrounds, temporal coordinate regression loss ( $\mathcal{L}_{tcr}$ ) for adjusting temporal coordinate of event proposal, and descriptiveness regression loss ( $\mathcal{L}_{des}$ ) for inferring the descriptive complexity of each proposal, which is defined as

$$\mathcal{L}_{TEP} = \mathcal{L}_{event} + \alpha \mathcal{L}_{tcr} + \beta \mathcal{L}_{des}, \quad (4)$$

where  $\alpha$  and  $\beta$  are tradeoff parameters. The event/background classification loss  $\mathcal{L}_{event}$  is measured as the standard softmax loss over the whole predicted proposal set  $\Phi_p$ . The temporal coordinate regression loss  $\mathcal{L}_{tcr}$  is devised as Smooth L1 loss [10] ( $S_{L1}$ ) between the positive predicted proposals and the corresponding ground truth proposals. Similar to [24], both of the temporal offsets for center location ( $\varphi_c$ ) of the predicted proposal and for its width ( $\varphi_w$ ) are regressed as

$$\mathcal{L}_{tcr} = \frac{1}{N_{pos}} \sum_{i=1}^{N_p} \mathbb{I}_i (S_{L1}(\varphi_c^i - g_c^i) + S_{L1}(\varphi_w^i - g_w^i)), \quad (5)$$

where  $\mathbb{I}_i$  denotes the assigned label for predicted proposal  $\phi_p^i$  (1 for positive sample and 0 for negative sample),  $N_{pos}$  is the number of positive samples,  $g_c^i$  and  $g_w^i$  represent the center location and width of ground truth proposal.

The descriptiveness regression loss  $\mathcal{L}_{des}$  is calculated as the Euclidean distance between the inferred descriptiveness score  $p_{des}^i$  of the predicted proposal and its sentence-level reward  $r(\mathcal{S}_i)$  in SG with regard to ground truth sentence:

$$\mathcal{L}_{des} = \frac{1}{N_p} \sum_{i=1}^{N_p} \left\| p_{des}^i - r(\mathcal{S}_i) \right\|_2^2. \quad (6)$$

In particular, for each positive predicted proposal  $\phi_p^i$ , we achieve its sentence-level reward  $r(\mathcal{S}_i)$  by directly feeding this proposal into SG module and comparing the generated sentence  $\mathcal{S}_i$  with the corresponding ground-truth sentence over evaluation metric (e.g., METEOR). For each negative sample, its sentence-level reward is naturally fixed as 0. Accordingly, by minimizing the descriptiveness regression loss, our prediction layer is additionally endowed with the ability to directly infer the approximate descriptiveness score (i.e., the sentence-level reward for captioning) of an event proposal without referring ground-truth sentences.

### 3.3. Sentence Generation

Given the set of selected predicted event proposals  $\Phi_{\hat{p}} \subset \Phi_p$  from TEP module, each proposal  $\phi_{\hat{p}} \in \Phi_{\hat{p}}$  is injected into attribute-augmented LSTM-based model in SG module for description generation. Specifically, the attributes representation  $\mathbf{A}$  of predicted proposal  $\phi_{\hat{p}}$  is firstly transformed into LSTM to inform the whole LSTM about the high-level attributes, followed by the proposal representation  $\mathbf{F}$  which is encoded into LSTM at the second time step. Then, LSTM decodes each output word based on previous word and previous step's LSTM hidden state.

**Descriptiveness-driven Temporal Attention.** One natural way to achieve the proposal representation  $\mathbf{F}$  is performing ‘‘mean pooling’’ process over all the clips within this proposal. However, in many cases, the generated description only relates to some key clips with low descriptive complexity. As a result, to pinpoint the local clips containing rich describable objects or scenes and further incorporate the contributions of different clips into producing proposal representation, a descriptiveness-driven temporal attention mechanism is employed on the predicted event proposal. Given the input proposal  $\phi_{\hat{p}}$  containing  $N_{\hat{p}_i}$  clips  $\{v_i\}_{i=1}^{N_{\hat{p}_i}}$ , the clip-level descriptiveness score  $p_{des}^{v_i}$  of each clip  $v_i$  is firstly achieved by holistically taking the average of all the descriptiveness scores of predicted proposals containing this clip  $v_i$ . Here we treat the clip-level descriptiveness score of each clip as one kind of temporal attention over all the clips within this proposals. Based on the attention distribution, we calculate the weighted sum of the clip features and obtain the aggregated proposal feature  $\mathbf{F}$  weighted by holistic attention score of each clip:

$$\alpha_{v_i} = p_{des}^{v_i} / \sum_{j=1}^{N_{\hat{p}_i}} p_{des}^{v_j}, \quad \mathbf{F} = \sum_{i=1}^{N_{\hat{p}_i}} \alpha_{v_i} \cdot \mathbf{f}_{v_i}, \quad (7)$$

where  $\alpha_{v_i}$  is the normalized attention score and  $\mathbf{f}_{v_i}$  is the clip representation of  $v_i$ . The aggregated proposal feature could be regarded as a more informative proposal feature since the most descriptive clips for sentence generation have been distilled with higher attention weights.

**Training.** The training objective in our SG module is formulated as the expected sentence-level reward loss in Eq.(2). Inspired from Self-critical Sequence Training (SCST) [25], the gradient of this objective is given by

$$\nabla_{\theta} \mathcal{L}_{SG} \approx -(r(\mathcal{S}^s) - r(\hat{\mathcal{S}})) \nabla_{\theta} \log p_{\theta}(\mathcal{S}^s), \quad (8)$$

where  $\mathcal{S}^s$  is a sampled sentence and  $r(\hat{\mathcal{S}})$  denotes the reward of baseline achieved by greedily decoding inference.

### 3.4. Joint Detection and Captioning

The overall objective of our dense video captioning is comprised of the training objective of TEP module in Eq.(4) and the reward loss of SG module in Eq.(2)

$$\mathcal{L} = \lambda_1 \mathcal{L}_{TEP} + \lambda_2 \mathcal{L}_{SG}, \quad (9)$$

where  $\lambda_1$  and  $\lambda_2$  are tradeoff parameters for TEP and SG, respectively. Note that descriptiveness inference could be regarded as a bridge which is not only leveraged in TEP for adjusting the event proposal from language perspective, but also integrated into SG for measuring the descriptiveness-driven temporal attention to boost sentence generation, enabling the interaction between TEP and SG. As a result, the overall objective function of our system can be solved through the joint and global optimization of detection and captioning in an end-to-end manner.

## 4. Experiments

We conduct our experiments on the ActivityNet Captions dataset [16] and evaluate our proposed system on both dense video captioning and temporal event proposal tasks.

### 4.1. Dataset

The dataset, ActivityNet Captions, is a recently collected large-scale dense video captioning benchmark, which contains 20,000 videos covering a wide range of complex human activities. Each video is aligned with a series of temporally annotated sentences. On average, there are 3.65 temporally localized sentences for each video, resulting in a total of 100,000 sentences. In our experiments, we follow the settings in [16], and take 10,024 videos for training, 4,926 for validation and 5,044 for testing.

### 4.2. Dense Video Captioning Task

We firstly investigate our system on dense video captioning task. The task is to detect individual events and then describe each event with natural language.

**Compared Approaches.** To empirically verify the merit of our proposed dense video captioning system, we compare the following video captioning baselines:

(1) Long Short-Term Memory (LSTM) [33]: LSTM utilizes a CNN plus RNN framework to directly translate from video pixels to natural language descriptions. The frame features are mean pooled to generate the video features.

(2) Sequence to Sequence - Video to Text (S2VT) [32]: S2VT incorporates both RGB and optical flow inputs, and the encoding and decoding of the inputs and word representations are learnt jointly in a parallel manner.

(3) Temporal Attention (TA) [34]: TA combines the frame representation from GoogleNet [29] and video clip representation from 3D CNN trained on hand-crafted descriptors. Furthermore, a soft attention mechanism is employed to dynamically attend to specific temporal regions of the video while generating sentences.

(4) Hierarchical Recurrent Neural Networks (H-RNN) [38]: H-RNN generates paragraphs by using one RNN to generate individual sentence and the second to capture the inter-sentence dependencies. Moreover, both spatial and temporal attention mechanisms are leveraged in H-RNN.

(5) Dense-Captioning Events (DCE) [16]: DCE leverages a multi-scale variant of DAPs [6] to localize temporal event proposals and S2VT [32] as base captioning module to describe each event. An attention module is further incorporated to exploit temporal context for dense captioning.

(6) Dense Video Captioning (DVC) is our complete system in this paper. Two slightly different settings of DVC are named as DVC-D and DVC-D-A. The former only incorporates the descriptiveness-driven temporal attention mechanism into LSTM in SG module and is trained with

out attributes and reinforcement learning based optimization, while the latter is more similar to DVC that only replaces the expected sentence-level reward loss in DVC with the traditional cross entropy loss.

Note that DCE is the only existing work on dense video captioning task and most previous video captioning works (e.g., LSTM, S2VT, TA, and H-RNN) focus on describing entire videos without detecting a series of events. Hence we compare the five video captioning baselines on dense video captioning task by feeding them with the fixed ground truth proposals or the learnt ones from our TEP module.

**Settings.** For video clip representation, we utilize the publicly available 500-way C3D in [16], whose dimension is reduced by PCA from the original 4,096-way output of *fc7* of C3D pre-trained on Sports-1M video dataset [13]. For representation of attributes/categories, we treat all the 200 categories on ActivityNet dataset [4] as the high-level semantic attributes and train the attribute detectors with cross entropy loss, resulting in the final 200-way vector of probabilities. Each word in the sentence is represented as “one-hot” vector (binary index vector in a vocabulary). The dimension of the input and hidden layers in LSTM are both set to 1,024. The tradeoff parameter  $\lambda_0$  leveraging the event probability and descriptiveness score for proposal selection is empirically set to 0.2. The tradeoff parameters  $\alpha$  and  $\beta$  in Eq.(4) are set as 0.5 and 10. For the tradeoff parameters  $\lambda_1$  and  $\lambda_2$  in Eq.(9), we set them as 1 and 20, respectively.

We mainly implement our DVC based on Caffe [12], which is one of widely adopted deep learning frameworks. The whole system is trained by Adam [14] optimizer. The initial learning rate is set as 0.00001 and the mini-batch size is set as 1. Note that SG module in our DVC is pre-trained with ground-truth proposal-sentence pairs. The sentence-level reward in SG is measured with METEOR.

**Evaluation Metrics.** For the evaluation of our proposed models, we follow the metrics in [16] to measure the ability to jointly localize and describe dense events. This metric computes the mean average precision (mAP) across tIoU thresholds of 0.3, 0.5, 0.7, and 0.9 when captioning the top 1,000 proposals. The precision of captions is measured by three evaluation metrics: BLEU@N [21], METEOR [1], and CIDEr-D [31]. All the metrics are computed by using the codes<sup>1</sup> released by ActivityNet Evaluation Server.

**Performance Comparison.** Table 1 shows the performances of different models on ActivityNet Captions validation set. Overall, the results across six evaluation metrics with ground truth proposals and learnt proposals consistently indicate that our proposed DVC achieves superior performances against other state-of-the-art video captioning techniques including non-attention models (LSTM, S2VT) and attention-based approaches (TA, H-RNN, DCE). In particular, the METEOR score of our DVC can achieve 10.33%

<sup>1</sup>[https://github.com/ranjaykrishna/densevid\\_eval](https://github.com/ranjaykrishna/densevid_eval)

Table 1. BLEU@N (B@N), METEOR (M) and CIDEr-D (C) scores of our DVC and other state-of-the-art video captioning methods for dense video captioning task on ActivityNet Captions validation set (left: performances with ground truth (GT) proposals; right: performances with the learnt proposals from our TEP module). All values are reported as percentage (%).

	with GT proposals						with learnt proposals					
	B@1	B@2	B@3	B@4	M	C	B@1	B@2	B@3	B@4	M	C
LSTM [33]	18.40	8.76	3.99	1.53	8.66	24.07	11.19	4.73	1.75	0.60	5.53	12.06
S2VT [32]	18.25	8.68	4.02	1.57	8.74	24.05	11.10	4.68	1.83	0.65	5.56	12.16
TA [34]	18.19	8.62	3.98	1.56	8.75	24.14	11.06	4.66	1.78	0.65	5.62	12.19
H-RNN [38]	18.41	8.80	4.08	1.59	8.81	24.17	11.21	4.79	1.90	0.70	5.68	12.35
DCE [16]	18.13	8.43	4.09	1.60	8.88	25.12	10.81	4.57	1.90	0.71	5.69	12.43
DVC-D	18.25	8.56	4.15	1.63	8.94	25.46	10.83	4.61	1.91	0.73	5.79	12.86
DVC-D-A	19.34	9.55	4.51	<b>1.71</b>	9.31	<b>26.26</b>	11.78	5.24	2.05	<b>0.74</b>	6.14	<b>13.21</b>
DVC	<b>19.57</b>	<b>9.90</b>	<b>4.55</b>	1.62	<b>10.33</b>	25.24	<b>12.22</b>	<b>5.72</b>	<b>2.27</b>	0.73	<b>6.93</b>	12.61

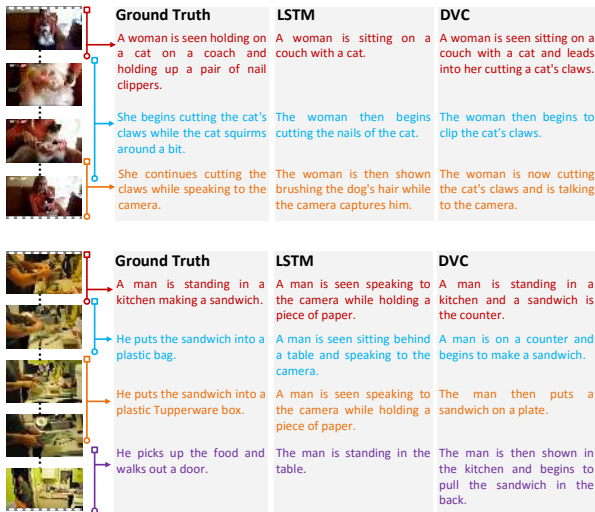


Figure 3. Dense video captioning results on ActivityNet Captions validation set. The output temporally localized sentences are generated by 1) Ground Truth, 2) LSTM, and 3) our DVC. We show the results with the highest overlap with ground truth captions.

with ground truth proposals, making the relative improvement over the best competitor DCE by 16.33%, which is considered as a significant progress on this benchmark. As expected, the METEOR score is dropped down to 6.93% when provided the predicted proposals from our TEP module instead of ground truth proposals. Moreover, DVC-D by additionally leveraging descriptiveness-driven temporal attention for dense video captioning, leads to a performance boost against LSTM. The result basically indicates the advantage of weighting each local clips with holistic attention score in producing proposal representation for sentence generation, instead of representing each proposal by directly performing “mean pooling” over its clips in LSTM. In addition, DVC-D-A by additionally augmenting LSTM with high-level semantic attributes, consistently improves DVC-D over all the metrics, but the METEOR scores are still lower than DVC. This confirms the effectiveness of utilizing reinforcement learning techniques for directly optimizing LSTM with METEOR-based reward loss, which harmonizes SG module with respect to its testing inference.

Figure 3 showcases a few dense video captioning results generated by different methods and human-annotated ground truth sentences. From these exemplar results, it is

Table 2. User study on two criteria: M1 - percentage of sets of captions generated by different methods that are evaluated as better/equal to human captions; M2 - percentage of sets of captions that pass Turing Test.

	Human	DVC	H-RNN [38]	TA [34]	S2VT [32]	LSTM [33]
M1	-	<b>35.1</b>	32.9	30.8	30.2	28.7
M2	<b>97.6</b>	<b>38.7</b>	36.3	34.6	34.1	31.9

easy to see that all of these automatic methods can generate somewhat relevant sentences, while our proposed DVC can generate more relevant and descriptive sentences by jointly exploiting descriptiveness-driven temporal attention mechanism and high-level attributes for boosting dense video captioning. For example, compared to phrase “brushing the dog’s hair” in the sentence generated by LSTM, “cutting the cat’s claws” in our DVC is more precise to describe the event proposal in the last proposal of the first video.

**Human Evaluation.** To better understand how satisfactory are the localized temporal event proposals and the corresponding generated sentences of different methods, we also conducted a human study to compare our DVC against four baselines, i.e., H-RNN, TA, S2VT, and LSTM. A total number of 12 evaluators from different education backgrounds are invited and a subset of 1K videos is randomly selected from validation set for the subjective evaluation. The evaluation process is as follows. All the evaluators are organized into two groups. We show the first group all the five sets of temporally localized sentences generated by each approach plus a series of temporally human-annotated sentences and ask them the question: Do the systems produce the set of temporally localized sentences resembling human-generated sentences? In contrast, we show the second group once only one set of temporally localized sentences generated by different approach or human annotations and they are asked: Can you determine whether the given set of sentences has been generated by a system or by a human being? From evaluators’ responses, we calculate two metrics: 1) M1: percentage of sets of captions that are evaluated as better or equal to human captions; 2) M2: percentage of sets of captions that pass the Turing Test. Table 2 lists the result of the user study. Overall, our DVC is clearly the winner for all two criteria. In particular, the percentage achieves 35.1% and 38.7% in terms of M1 and M2, respectively, making the absolute improvement over the best competitor H-RNN by 2.2% and 2.4%.

Table 3. Leaderboard of the published state-of-the-art dense video captioning models on the online ActivityNet evaluation server.

	DVC (P3D)	DVC (C3D)	TAC [9]	DCE [16]
METEOR	<b>12.96</b>	<b>10.36</b>	9.61	4.82

**Performance on ActivityNet Evaluation Server.** We also submitted our best run in terms of METEOR score, i.e., DVC, to online ActivityNet evaluation server and evaluated the performance on official testing set. Table 3 shows the performance Leaderboard on official testing set. Please note that here we design two submission runs for our DVC, i.e., DVC (C3D) and DVC (P3D). The input clip features of the two runs are 500-way C3D features and 2048-way output of *pool5* layer from P3D ResNet [22], respectively. Compared to the top performing methods, our proposed DVC (C3D) achieves the best METEOR score. In addition, when leveraging the clip feature from P3D ResNet, our METEOR score on testing set is further boosted up to 12.96%, ranking the first on the Leaderboard.

### 4.3. Temporal Event Proposal Task

The second experiment is conducted on temporal event proposal task, which evaluates our TEP module’s capability to adequately localize all events for a given video.

**Compared Approaches.** We compare our DVC with three state-of-the-art temporal action proposal methods:

(1) Temporal Actionness Grouping (TAG) [39]. TAG utilizes actionness classifier to generate actionness curve, followed by the watershed algorithm to produce basins. The proposals are finally generated by grouping the basins.

(2) Dense-Captioning Events (DCE) [16]. DCE leverages a multi-scale variant of LSTM-based action proposal model in [6] to localize temporal event proposals.

(3) Temporal Unit Regression Network (TURN) [8]. TURN jointly predicts action proposals and refines the temporal boundaries by temporal coordinate regression.

**Evaluation Metrics.** For temporal action proposal task, we adopt the Area-Under-the-Curve (AUC) score for Average Recall vs. Average Number of Proposals per Video (AR-AN) curve in [9] as the evaluation metric. AR is defined as the mean of all recall values using tIoU thresholds between 0.5 and 0.95 (step size: 0.05), and AN denotes the total number of proposals divided by the number of videos.

**Performance Comparison.** Figure 4 shows the AR-AN curves of four runs on ActivityNet Captions validation set for temporal event proposal task. Overall, the quantitative results with regard to AUC score indicate that our DVC outperforms other methods. In particular, by leveraging temporal coordinate regression for adjusting temporal boundary of detected proposals, DCE and TURN lead a large performance boost against TAG. Moreover, DVC by additionally incorporating descriptiveness regression into TEP module further improves DCE and TURN. The result indicates the advantage of joint detection and captioning, which refines the event proposal from language perspective.

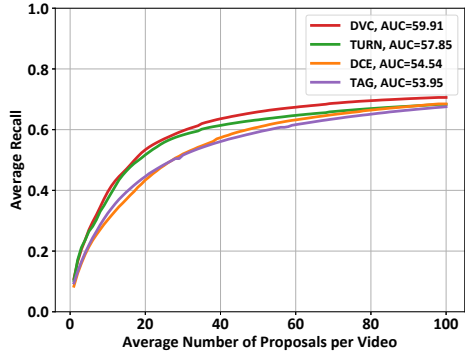


Figure 4. The AR-AN curves of different approaches on ActivityNet Captions validation set for temporal event proposal task.

Table 4. Effects of utilizing multiple anchor layers.

<i>conv</i> <sub>3~8</sub>	<i>conv</i> <sub>9</sub>	<i>conv</i> <sub>10</sub>	<i>conv</i> <sub>11</sub>	<i>conv</i> <sub>12</sub>	AUC	Parameter Number
✓					55.73	16.5M
✓	✓				57.38	18.9M
✓	✓	✓			58.75	21.2M
✓	✓	✓	✓		59.91	23.6M
✓	✓	✓	✓	✓	60.07	26.0M

**Effects of Multiple Anchor Layers.** In order to show the relationship between performance and the number of anchor layers with different temporal resolutions, we progressively stack anchor layers with decreasing temporal resolutions and compare the performances. The results shown in Table 4 indicate increasing the number of anchor layers with different temporal resolutions can generally lead to performance improvements. Meanwhile, the number of parameters in all adopted anchor layers increases. Thus, we finally adopt *conv*<sub>3~11</sub> as anchor layers as that has a better tradeoff between performance and model complexity.

## 5. Conclusions

We have presented a novel deep architecture which unifies the temporal localization of event proposals and sentence generation for dense video captioning. Particularly, we study the problems of how to build the interaction across the two sub challenges (i.e., temporal event proposal and sentence generation) and how to integrate such interaction into a deep learning framework for enhancing dense video captioning. To verify our claim, we have devised a descriptiveness regression component and incorporated it into a single shot detection structure, on one hand to adjust the event proposal from language perspective in TEP module, and on the other, to measure the descriptive complexity of each event in SG module. Experiments conducted on ActivityNet Captions dataset validate our model and analysis. More remarkably, we achieve superior results over state-of-the-art methods when evaluating our framework on both dense video captioning and temporal event proposal tasks.

**Acknowledgments.** This work was supported in part by NSFC under Grant 61672548, U1611461, and the Guangzhou Science and Technology Program, China, under Grant 201510010165.



## References

- [1] S. Banerjee and A. Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *ACL workshop*, 2005.
- [2] S. Buch, V. Escorcia, C. Shen, B. Ghanem, and J. C. Niebles. Sst: Single-stream temporal action proposals. In *CVPR*, 2017.
- [3] F. Caba Heilbron, J. Carlos Niebles, and B. Ghanem. Fast temporal activity proposals for efficient detection of human actions in untrimmed videos. In *CVPR*, 2016.
- [4] F. Caba Heilbron, V. Escorcia, B. Ghanem, and J. Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, 2015.
- [5] O. Duchenne, I. Laptev, J. Sivic, F. Bach, and J. Ponce. Automatic annotation of human actions in video. In *ICCV*, 2009.
- [6] V. Escorcia, F. C. Heilbron, J. C. Niebles, and B. Ghanem. Daps: Deep action proposals for action understanding. In *ECCV*, 2016.
- [7] H. Fang, S. Gupta, et al. From captions to visual concepts and back. In *CVPR*, 2015.
- [8] J. Gao, Z. Yang, C. Sun, K. Chen, and R. Nevatia. Turn tap: Temporal unit regression network for temporal action proposals. In *ICCV*, 2017.
- [9] B. Ghanem, J. C. Niebles, C. Snoek, F. C. Heilbron, H. Alwassel, R. Khrisna, V. Escorcia, K. Hata, and S. Buch. Activitynet challenge 2017 summary. *arXiv preprint arXiv:1710.08011*, 2017.
- [10] R. Girshick. Fast r-cnn. In *ICCV*, 2015.
- [11] S. Guadarrama, N. Krishnamoorthy, G. Malkarnenkar, S. Venugopalan, R. Mooney, T. Darrell, and K. Saenko. Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition. In *ICCV*, 2013.
- [12] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *MM*, 2014.
- [13] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014.
- [14] D. Kingma and J. Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.
- [15] A. Kojima, T. Tamura, and K. Fukunaga. Natural language description of human activities from video images based on concept hierarchy of actions. *IJCV*, 2002.
- [16] R. Krishna, K. Hata, F. Ren, L. Fei-Fei, and J. C. Niebles. Dense-captioning events in videos. In *ICCV*, 2017.
- [17] T. Lin, X. Zhao, and Z. Shou. Single shot temporal action detection. In *MM*, 2017.
- [18] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. SSD: Single shot multibox detector. In *ECCV*, 2016.
- [19] Y. Pan, T. Mei, T. Yao, H. Li, and Y. Rui. Jointly modeling embedding and translation to bridge video and language. In *CVPR*, 2016.
- [20] Y. Pan, T. Yao, H. Li, and T. Mei. Video captioning with transferred semantic attributes. In *CVPR*, 2017.
- [21] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, 2002.
- [22] Z. Qiu, T. Yao, and T. Mei. Learning spatio-temporal representation with pseudo-3d residual networks. In *ICCV*, 2017.
- [23] M. Ranzato, S. Chopra, M. Auli, and W. Zaremba. Sequence level training with recurrent neural networks. In *ICLR*, 2016.
- [24] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015.
- [25] S. J. Rennie, E. Marcheret, Y. Mroueh, J. Ross, and V. Goel. Self-critical sequence training for image captioning. In *CVPR*, 2017.
- [26] M. Rohrbach, W. Qiu, I. Titov, S. Thater, M. Pinkal, and B. Schiele. Translating video content to natural language descriptions. In *ICCV*, 2013.
- [27] Z. Shou, D. Wang, and S.-F. Chang. Temporal action localization in untrimmed videos via multi-stage cnns. In *CVPR*, 2016.
- [28] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *NIPS*, 2014.
- [29] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, 2015.
- [30] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, 2015.
- [31] R. Vedantam, C. Lawrence Zitnick, and D. Parikh. Cider: Consensus-based image description evaluation. In *CVPR*, 2015.
- [32] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko. Sequence to sequence - video to text. In *ICCV*, 2015.
- [33] S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko. Translating videos to natural language using deep recurrent neural networks. In *NAACL HLT*, 2015.
- [34] L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, and A. Courville. Describing videos by exploiting temporal structure. In *ICCV*, 2015.
- [35] T. Yao, Y. Li, Z. Qiu, F. Long, Y. Pan, D. Li, and T. Mei. MSR Asia MSM at activitynet challenge 2017: Trimmed action recognition, temporal action proposals and dense-captioning events in videos. In *CVPR ActivityNet Challenge Workshop*, 2017.
- [36] T. Yao, Y. Pan, Y. Li, and T. Mei. Incorporating copying mechanism in image captioning for learning novel objects. In *CVPR*, 2017.
- [37] T. Yao, Y. Pan, Y. Li, Z. Qiu, and T. Mei. Boosting image captioning with attributes. In *ICCV*, 2017.
- [38] H. Yu, J. Wang, Z. Huang, Y. Yang, and W. Xu. Video paragraph captioning using hierarchical recurrent neural networks. In *CVPR*, 2016.
- [39] Y. Zhao, Y. Xiong, L. Wang, Z. Wu, X. Tang, and D. Lin. Temporal action detection with structured segment networks. *ICCV*, 2017.