Towards Diverse Paragraph Captioning for Untrimmed Videos

Yuqing Song¹; Shizhe Chen²; Qin Jin⁠†
¹Renmin University of China, ²INRIA
syuqing@ruc.edu.cn, cshizhe@gmail.com, qjin@ruc.edu.cn

Abstract

Video paragraph captioning aims to describe multiple events in untrimmed videos with descriptive paragraphs. Existing approaches mainly solve the problem in two steps: event detection and then event captioning. Such two-step manner makes the quality of generated paragraphs highly dependent on the accuracy of event proposal detection which is already a challenging task. In this paper, we propose a paragraph captioning model which eschews the problematic event detection stage and directly generates paragraphs for untrimmed videos. To describe coherent and diverse events, we propose to enhance the conventional temporal attention with dynamic video memories, which progressively exposes new video features and suppresses over-accessed video contents to control visual focuses of the model. In addition, a diversity-driven training strategy is proposed to improve diversity of paragraph on the language perspective. Considering that untrimmed videos generally contain massive but redundant frames, we further augment the video encoder with keyframe awareness to improve efficiency. Experimental results on the ActivityNet and Charades datasets show that our proposed model significantly outperforms the state-of-the-art performance on both accuracy and diversity metrics without using any event boundary annotations. Code will be released at https://github.com/syuqings/video-paragraph.

1. Introduction

Describing videos with natural language sentences, a.k.a. video captioning, has attracted increasing research attentions due to the rapid emergence of videos in our lives. The dominant video captioning task [19, 46, 31, 42, 45] focuses on generating a single sentence to describe a carefully trimmed video which mainly contains one major event within short duration such as 10-20 seconds [40, 37]. However, the videos in the wild are mostly untrimmed with rich temporal event structures. A single sentence is insufficient to convey fine-grained information in such untrimmed videos. Therefore, recent works [39, 21, 13] have attempted to generate a story-oriented paragraph with multiple sentences to comprehensively describe video contents.

Existing works [39, 21, 13] mainly adopt a two-stage framework for video paragraph captioning: firstly detecting event segments in the video, and then generating the event description for each segment. Despite being reasonable, the framework requires temporal segment coordinates for descriptions in the paragraph to train the model, which are expensive to annotate. Moreover, since event categories are extremely diverse in open-domain untrimmed videos, it is quite challenging to detect precise event segments compared with the action detection task [25, 47, 4], which has a fixed category list. The poorly detected events greatly harm the performance of paragraph captioning in existing frameworks. As a result, several works [21, 13] use ground-truth event segments to generate video paragraphs, which cannot generalize to videos without such event annotations.

However, is event detection really necessary for video paragraph captioning? Let’s review a simpler task of image paragraph captioning. The state-of-the-art approaches [16, 17] directly generate sentences from images without predicting sequences of image coordinates. The generated paragraphs have shown good capability to capture descriptive logic such as from foreground to background. Motivated by these works, we aim to eschew the costly event segment detection process, and efficiently generate video paragraph descriptions in a single stage.

Compared with the image counterpart, there are mainly three challenges for video paragraph captioning when event segments are unavailable. Firstly, an untrimmed video generally consists of hundreds or thousands of frames, while an image contains much fewer region candidates to be attended to. Therefore, it consumes more computation resources during description generation. Secondly, the large number of frame candidates also makes it hard for the captioning model to learn an effective attention mechanism to form a coherent descriptive logic and describe diverse events in the video, especially when the training examples are limited.

---

¹Equal contribution. This work was performed when Shizhe Chen was at Renmin University of China.
⁠†Corresponding author.
Thirdly, the captioning model usually tends to generate redundant words and phrases that are of high frequency in the dataset especially for the long paragraph generation.

In this work, we propose an one-stage framework to tackle the above challenges for diverse and efficient video paragraph generation. Considering that there are many redundant frames in untrimmed videos, we propose to automatically select keyframes during the video encoding via additional video semantic summary loss and sparsity loss. In this way, only keyframes are used to generate the long paragraph during inference, which improves the computational efficiency. To guide the model in effective description logic learning for diverse and coherent events, we propose to improve conventional temporal attention with dynamic video memory which tracks and controls visual focuses in the video. It includes an “add” operation to progressively expose new video frames to the model, and an “erase” operation to suppress over-accessed video contents. To further improve diversity of generated paragraphs from language perspective, we improve the training objective with token-level and sequence-level high-frequency penalties to encourage generating more unique expressions. Experiments show that our model outperforms two-stage methods which even utilize ground-truth event segments on ActivityNet dataset, and also achieves the state-of-the-art result on Charades dataset which does not have temporal annotations.

The main contributions of this work are as follows:

- To the best of our knowledge, we are the first to eschew event detection stage and directly generate paragraphs for untrimmed videos, which avoids the dependence on expensive event temporal annotations.
- We propose an attention mechanism with dynamic video memories and diversity-driven training objectives to generate coherent and diverse paragraph from video and language perspectives, and improve generation efficiency via keyframe-aware video encoder.
- Our model achieves state-of-the-art results on both ActivityNet and Charades datasets without using any event boundary annotations.

2. Background: Vanilla Paragraph Captioning

Given an untrimmed video \( v \), the video paragraph captioning task aims to generate a paragraph \( y = \{y_1, \ldots, y_T\} \) to describe events in \( v \), where \( y_t \) denotes the \( t \)-th word in the paragraph. In the following, we introduce a vanilla video paragraph captioning model without the event segment detection stage and discuss its limitations.

The vanilla model is similar to conventional video captioning models [19, 46, 31] based on the encoder-decoder framework [27]. The encoder transforms \( v \) into a sequence of clip-level features. Specifically, we first divide \( v \) into non-overlapping clips with 64 frames per clip and use pre-trained CNNs [10, 3] to extract features for each clip as \( \mathcal{X}^0 = \{x_1^0, \ldots, x_L^0\} \), where \( L \) is the number of clips. To encode long-range temporal dependencies among clips, we apply \( N \) transformer layers on \( \mathcal{X}^0 \) as follows:

\[
\mathcal{X}^i = \text{FFN}(\mathcal{X}^{i-1} + \text{MultiHead}(\mathcal{X}^{i-1}, \mathcal{X}^{i-1}, \mathcal{X}^{i-1}))
\]

where FFN(\( \cdot \)) and MultiHead(\( \cdot \)) denote feed-forward network and multi-head attention as in [29]. The hidden state \( \mathcal{X}^N \) is used as the encoded video feature \( \mathcal{Y}^{\text{enc}} \in \mathbb{R}^{L \times d} \), where \( d \) is the feature dimension.

For the decoder, we use \( N \) layers of transformer due to the advantage of its structure in long text generation [50, 13]. Besides the self-attention as in the encoder, the decoder further adopts the cross-modal multi-head attention to compute attention weights on \( \mathcal{Y}^{\text{enc}} \) at each decoding step. Therefore, each word is generated conditioning on previously predicted words and the attended video contents.

The captioning model is typically trained by maximum likelihood estimation (MLE) given the ground-truth pair \( (v, y^*) \), where \( y^* = \{y_1^*, \ldots, y_T^*\} \), which is:

\[
\mathcal{L}_{\text{mle}} = -\frac{1}{T} \sum_{t=1}^{T} \log p(y_t^* | y_{<t}^*, v)
\]

To address the exposure bias and target mismatch [23] problems in MLE, reinforcement learning (RL) [22] is usually adopted to further improve the model with sequence-level non-differentiable caption rewards as follows:

\[
\mathcal{L}_{\text{rl}} = -\frac{1}{T} \sum_{t=1}^{T} r(y_t^*) \log p(y_t^* | y_{<t}^*, v)
\]

where \( y^* = \{y_1^*, \ldots, y_T^*\} \) is a paragraph sampled from the model and \( r(\cdot) \) is the reward function.

Without event segment annotations, the above vanilla model suffers from three limitations for video paragraph captioning. Firstly, the untrimmed video usually contains a large number of clips, while the vanilla encoder feeds all clip features to the following decoder which brings huge attention computation burden for long paragraph generation. Secondly, due to the large amount of clip candidates and limited training examples, it is hard for the decoder to learn effective attention mechanism to form coherent descriptive logic. Finally, both MLE and RL training make the model more likely to generate high-frequency words and phrases, and thus harm the diversity of generated paragraphs. Therefore, it is essential to address these limitations to make the one-stage model more practical.

3. The Proposed Method

In this section, we introduce the proposed video paragraph captioning model, which is illustrated in Figure 1.
We first describe the keyframe-aware video encoder in Section 3.1, which selects key frames during video encoding to improve attention computation efficiency in decoding. Then in Section 3.2, we present a more effective cross-modal attention mechanism enhanced by dynamic video memories, including progressive memory exposure to guide the model describing temporally coherent events, and over-accessed memory decay to reduce repetitions on described events. Finally, we present the proposed diversity-driven training objective in Section 3.3, which improves the language diversity via penalizing tokens and phrases of high-frequency.

3.1. Keyframe-aware Video Encoder

Considering that there is a large amount of clips in untrimmed videos which brings huge attention computation burden to the decoder, we propose to explicitly select key frames during video encoding. We augment the vanilla encoder with a keyframe selection layer, which predicts the informativeness of each clip based on its contextual representation. In the $i$-th encoding layer, the encoded video feature is computed as follows:

$$\hat{X}_i = X_{i-1} + \text{MultiHead}(X_{i-1}, X_{i-1}, X_{i-1})$$ \hspace{1cm} (4)

$$s^i = \sigma(\text{FFN} (\hat{X}_i))$$ \hspace{1cm} (5)

$$X_i = \hat{X}_i \cdot s^i$$ \hspace{1cm} (6)

where $\sigma$ is the sigmoid function, $s_{j}^i$ is a scalar to infer the informativeness of the $j$-th clip to the $i$-th encoding layer. In this way, we can employ $s^N \in \mathbb{R}^L$ to identify the key frames to be used in the following decoder.

However, the paragraph generation loss alone cannot provide sufficient supervision for the keyframe selection and the poorly selected video features can hinder effective attention learning in the decoder. Since key frames can well represent the semantic contents of a whole video, they are expected to reconstruct original video in the semantic space. Therefore, we propose to reconstruct high-level semantic embedding of the video in a visual-semantic joint embedding space. Specifically, we utilize the video-text retrieval [9] task as a proxy task to pre-train such visual-semantic joint embedding space. We first feed the video feature sequence $X^0$ and the ground-truth paragraph $y^*$ to GRUs [6] respectively to get a global encoding vector for each of them. Then they are mapped to the joint embedding space with the hard negative triplet loss [9] to ensure the video/text with similar semantics will be embeded closer.

After pre-training, we fix the video GRU and employ it to compute the video reconstruction loss as follows:

$$\mathcal{L}_{\text{reconst}} = \left\| \text{GRU}^v (X_{key}^0) - \text{GRU}^v (X^0) \right\|_2$$ \hspace{1cm} (7)

where $X_{key}^0 = X^0 \cdot s^N$, which is the soft selected keyframe features. We choose to reconstruct $X^0$ rather than $y^*$ because there is less cross-modal gap between $X_{key}^0$ and $X^0$ making the learning more effective. To penalize a large number of key frames being selected, we further introduce a sparsity loss as follows:

$$\mathcal{L}_{\text{sparsity}} = \frac{1}{L} \sum_{j=1}^{L} s_{j}^N - \delta$$ \hspace{1cm} (8)

where $L$ is the total number of video clips and $\delta$ is the hyperparameter which denotes the selecting ratio of key frames. Notice that in the training phase, we use the soft selection of key frames for the gradient back-propagation, while at inference time, we use $s^N$ to select the top $\lceil \delta L \rceil$ key frames to reduce the computational cost for decoding efficiency.

3.2. Attention with Dynamic Video Memories

With the encoded video feature sequence $y^{enc}$, the decoder employs temporal attention mechanism to generate
the paragraph. However, the video paragraphs usually contain rich temporal logic structures, which are hard to learn by traditional attentions from limited training examples. Therefore, we enhance the temporal attention mechanism in the decoder with dynamic video memories.

Instead of attending on the same $\mathcal{V}^{enc}$ at each decoding step, our model attends on video memory $\mathcal{M}_t = \{m_{t,1}, \cdots, m_{t,L}\}$ at each step $t$, which are dynamically updated to make visual attentions moving coherently on diverse events. Suppose $\alpha_t \in \mathbb{R}^L$ (averaged from multiple heads and layers) is the overall attention weights over the video memory $\mathcal{M}_t$ at $t$-th step. We utilize attention histories $\{\alpha_{t-W}, \cdots, \alpha_t\}$ with a history window $W$ to update $\mathcal{M}_t$ to $\mathcal{M}_{t+1}$ for the attention at next step. We use the attention histories instead of $\alpha_t$ because we expect to update the video memory when a complete phrase or sentence has been generated. The attention weights for a single word are noisy to indicate if the model should move to other frames. We aggregate the attention histories into $\tilde{\alpha}_t \in \mathbb{R}^L$ as follows to make attentions from more recent steps more important:

$$\tilde{\alpha}_t = \sum_{j=0}^W w_j \alpha_{t-j}$$  \hfill (9)

$$w_j = \frac{e^{1-(j/W)}}{\sum_{k=0}^W e^{1-(k/W)}}$$  \hfill (10)

where $w_j$ is the history decay weight. The $\tilde{\alpha}_t$ is then used to update the video memory via two operations, which include an “add” operation in progressive memory exposure to progressively add more video clip features to the memory and an “erase” operation in over-access memory decay to erase already described clips.

**Progressive Memory Exposure.** To keep the event description coherent such as following the temporal order, we propose to progressively expose the video feature sequence $\mathcal{V}^{enc}$ to the attended video memory $\mathcal{M}$. We first initialize the $\mathcal{M}_0$ as follows:

$$\mathcal{M}_0 = u_0 \cdot \mathcal{V}^{enc}$$  \hfill (11)

$$u_{0,i} = \begin{cases} 1 - (i/S), & i \leq S \\ 0, & i > S \end{cases}$$  \hfill (12)

where $u_i \in \mathbb{R}^L$ denotes the exposure status at step $t$, which records the proportion of each clip feature added to the video memory. The $u_{t,i} \in [0, 1]$ is constantly updated, where $u_{t,i} = 1$ indicates that the $i$-th clip feature should not be added anymore. $S$ is the initialization window length. It can make the decoder focus on the beginning of video first, rather than randomly starting the paragraph generation.

We propose an adding gate $g^a_t \in [0, 1]$ to determine whether we should “add” new features to the memory at step $t$. Because when non-visual words are generated or the accessed video frame has not been fully described, the video memories should be updated less. The gate $g^a_t$ is computed as follows:

$$g^a_t = \sigma(f_{add}(h_t; \theta_{add}))$$  \hfill (13)

where $\sigma$ is the sigmoid function, $h_t$ is the output hidden state at the $t$-th step and $f_{add}$ is a fully connected network parameterized by $\theta_{add}$.

Then we compute for each clip feature whether it needs to be added in the new video memory according to their visual relevance with previous context $\hat{c}_t$. The context can make the model describe events which are relevant to the previous one at next step to keep the event coherence. The $\hat{c}_t$ is computed based on aggregated attention history $\tilde{\alpha}_t$ as follows:

$$\hat{c}_t = \sum_{i=1}^L \tilde{\alpha}_{t,i} \cdot v^c_{i}$$  \hfill (14)

Therefore, the probabilities of each clip feature to be added to the video memory is computed as:

$$p^a_{t,i} = \sigma(f_{vis}([v^c_{i}; \hat{c}_t]; \theta_{vis}))$$  \hfill (15)

where $f_{vis}$ is the fully connected network similar to $f_{add}$.

Based on the adding gate $g^a_t$ and adding probability $p^a_{t,i}$ of each clip feature, we gradually add video features to the memories, which is:

$$\hat{m}_{t+1,i} = m_{t,i} + g^a_t (1 - u_{t,i}) p^a_{t,i} v^c_{i}$$ \hfill (16)

$$u_{t+1,i} = u_{t,i} + g^a_t (1 - u_{t,i}) p^a_{t,i}$$ \hfill (17)

where $\hat{m}_{t+1,i}$ is the intermediate memory feature which will be further processed with the “erase” operation.

**Over-accessed Memory Decay.** In addition to the description coherence, describing diverse content of the video is also important to the video paragraph generation. To prevent the decoder from only focusing on a few salient frames, we propose to weaken the already accessed features to encourage the model to describe more unseen video frames. Similar to the progressive memory exposure mechanism, we employ $g^e_t$ as an erasing gate to determine whether to erase the memory at the $t$-th step as follows:

$$g^e_t = \sigma(f_{ers}(h_t; \theta_{ers}))$$  \hfill (18)

Considering that the attention weights can indicate the access intensity of each clip feature, we update the video memory with the guidance of that. Besides the attention weights, to ensure the highly attended features to be erased have actually been described, we further compute their semantic relevance to the generated words as follows:

$$\hat{h}_t = \sum_{j=0}^W w_j h_{t-j}$$  \hfill (19)

$$p^e_{t,i} = \sigma(f_{sem}([\hat{m}_{t+1,i}; \hat{h}_t]; \theta_{sem}))$$  \hfill (20)

where $\hat{h}_t$ is the history hidden states computed similar to $\tilde{\alpha}_t$ in (9). Finally, the video memories can be updated to $\mathcal{M}_{t+1}$ as follows:

$$m_{t+1,i} = m_{t+1,i} (1 - g^e_t \tilde{\alpha}_{t,i} p^e_{t,i})$$  \hfill (21)
3.3. Diversity-driven Training

The dynamic video memories can help the model to describe diverse video content, while diverse language expression is also essential to the paragraph generation. The typical MLE and RL training objectives for the captioning model both force the model to fit the ground-truth distribution, which makes the decoder prefer high-frequency tokens and phrases. It not only results in dull and repetitive expressions, but also makes the model generate wrong descriptions, regardless of video content. Motivated by the unlikelihood training [38], we improve the training objectives with token- and phrase-level high-frequency penalties.

Token-level Training. In the token-level training, we augment the MLE objective with high-frequency word penalties as [38]. Considering that the model tends to repeat words that have been generated before, we define the previous context words as the high-frequency tokens for the current training pair (v, y∗). Therefore, the MLE loss function (2) is changed to:

$$L_{\text{mle}} = - \frac{1}{T} \sum_{t=1}^{T} \left( \log p(y_t^s|y_{<t}^s,v) + \sum_{c \in \mathcal{C}^t} \log(1-p(c|y_{<t}^s,v)) \right)$$ (22)

where $\mathcal{C}^t = \{y_1^s, \ldots, y_{t-1}^s\} \setminus \{y_t^s\}$ is the candidate word set to be penalized. In this way, not only the probabilities of ground-truth words are enhanced, but also the probabilities of wrong candidates with high-frequency are penalized.

Sequence-level Training. In the sequence-level training, we introduce the phrase-level penalty to the RL loss function (3). Specifically, we compute the $n$-gram frequency of the training annotations to create a phrase frequency look-up table. The inverse document frequency (idf) score can represent the uniqueness of n-grams, which is employed as the diversity reward in reinforcement learning. To avoid the model generating meaningless phrases which of course have low frequencies in the annotations, we combine the diversity reward with the relevance reward computed by CIDEr [30]. The RL loss function (3) is changed to:

$$L_{\text{rl}} = - \frac{1}{T} \sum_{t=1}^{T} \left( r_{\text{rl}}(y_t^s) + \beta r_{\text{div}}(y_t^s) \right) \log p(y_t^s|y_{<t}^s,v)$$

$$r_{\text{div}}(y_t^s) = \frac{1}{k} \sum_{ ph \in H_n(y_t^s) } \frac{1}{\text{freq}(ph)}$$ (23)

where $H_n(y_t^s)$ is the set of $n$-grams containing $y_t^s$ in $y^s$, and $k$ is the size of $H_n(y_t^s)$. $\beta$ is the hyper-parameter to balance the diversity and accuracy. In practice, we normalize both $r_{\text{rl}}(\cdot)$ and $r_{\text{div}}(\cdot)$ with baseline rewards of sentences sampled by greedy search as in [23] for training stability.

We first train the whole model with $L_{\text{mle}}$ and video summary losses as follows:

$$L = L_{\text{mle}} + \lambda_1 L_{\text{reconst}} + \lambda_2 L_{\text{sparsity}}$$ (25)

where $\lambda_1$ and $\lambda_2$ are hyper-parameters. Then we fine-tune the model with $L_{\text{rl}}$ in reinforcement learning.

4. Experiments
4.1. Datasets and Experimental Settings

Datasets. We conduct experiments on the ActivityNet Captions dataset [12] and Charades Captions dataset [36]. ActivityNet Captions dataset contains 10,009 videos for training, 4,917 for validation and 5,044 for testing. Each video in the training set has a single reference paragraph while each video in the validation set has two reference paragraphs. Since the test set is held for the challenge evaluation, we follow previous works [13, 49] to split the validation set into two subsets: ae-val with 2,460 videos for validation and ae-test with 2,457 videos for test. Charades Captions dataset is processed from the Charades dataset [26], which contains 6,963 videos for training, 500 for validation and 1,760 for testing. Each video is annotated with multiple paragraphs.

Evaluation Metrics. We evaluate the paragraph generation qualities from two aspects, accuracy and diversity respectively. For the accuracy measurement, we evaluate the generated paragraph against the ground-truth with three standard metrics as [15, 21, 39], including BLEU@4 [20], METEOR [7] and CIDEr [30]. Since the standard metrics do not consider much about diversity of the paragraph, we further evaluate the generated paragraphs with diversity metrics. Following [21], we evaluate the diversity using two types of metrics: 1) $n$-gram diversity (Div@$n$) [24]: the ratio of unique $n$-grams to the total number of words in the paragraph, which is widely used for diversity evaluation; and 2) $n$-gram repetition (Rep@$n$) [39]: the ratio of $n$-gram repetitions to the total number of $n$-grams.

Implementation Details. For the videos, we use ResNet-200 [10] pretrained on ImageNet and I3D (RGB+Flow) [3] pretrained on Kinetics dataset to extract clip-level features of dimensionality 4096D. We truncate video clips with maximum number of 150. For the texts in ActivityNet dataset, we truncate the paragraph with maximum length of 150 and build the vocabulary with 10,246 words. For the Charades dataset, we truncate the paragraph with maximum length of 100 and build the vocabulary with 2,692 words. We set the number of encoder and decoder layers as $N = 3$, the hidden size as $d = 512$ and the number of attention heads as 8. The start window length $S$ in Eq.(12) is set as 50, according to dataset statistics that the first 1/3 length of the video are likely to belong to the first event in the paragraph description. We set the history window length $W = 20$ in Eq.(9) based on the average length of a single sentence. For the phrase penalty in Eq.(24), we set the $n$ to 4. The $\beta$ in Eq.(23) is set to 0.3, and the weights of video summary losses in Eq.(25) are set as $\lambda_1 = \lambda_2 = 0.5$. During training, we use the label smoothing [28] with value set...
Table 1. Comparison with state-of-the-art approaches for video paragraph generation on ActivityNet Captions ae-test split. “Train” and “Infer” indicate if the video segment annotations are needed at training and inference time.

<table>
<thead>
<tr>
<th>#</th>
<th>Methods</th>
<th>Segment Annotation</th>
<th>Accuracy</th>
<th>Diversity</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Train Infer</td>
<td>BLEU@4</td>
<td>METEOR</td>
</tr>
<tr>
<td>1</td>
<td>MFT [39]</td>
<td>✓ ✓</td>
<td>10.33</td>
<td>15.09</td>
</tr>
<tr>
<td>2</td>
<td>Transformer† [50]</td>
<td>✓ ✓</td>
<td>10.38</td>
<td>16.33</td>
</tr>
<tr>
<td>3</td>
<td>AdvInf‡ [21]</td>
<td>✓ ✓</td>
<td>10.89</td>
<td>17.41</td>
</tr>
<tr>
<td>4</td>
<td>MART† [13]</td>
<td>✓ ✓</td>
<td>10.54</td>
<td>17.12</td>
</tr>
</tbody>
</table>

| 5 | MFT [39] | ✓ x | 8.45 | 14.75 | 14.15 | - | - | 17.59 |
| 6 | Vanilla | x x | 11.53 | 15.91 | 24.11 | 64.92 | 82.34 | 3.17 |
| 7 | Ours | x x | 12.20 | 16.10 | 27.36 | 68.33 | 84.26 | 2.63 |

| 8 | Human | - - | - - | - - | 68.60 | 85.40 | 0.83 |

Table 2. Captioning results on Charades Captions dataset.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Accuracy</th>
<th>Diversity</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>B@4 M C</td>
<td>D@1 D@2 R@4</td>
</tr>
<tr>
<td>HRL [36]</td>
<td>18.80 19.50 23.20</td>
<td>- - -</td>
</tr>
<tr>
<td>Vanilla</td>
<td>19.19 19.80 25.30</td>
<td>72.90 86.13 1.23</td>
</tr>
<tr>
<td>Ours</td>
<td>20.34 20.05 27.54</td>
<td>76.18 87.31 0.92</td>
</tr>
<tr>
<td>Human</td>
<td>- - -</td>
<td>79.90 90.81 0.10</td>
</tr>
</tbody>
</table>

4.2. Comparison with the State-of-the-arts

We compare our model with the following state-of-the-art methods, which all use event segments (either ground-truth events or automatically generated events) for video paragraph generation.

- **MFT [39]**: A LSTM based model which couples two RNNs for the event detection and event captioning respectively. The previously detected events and generated captions are exploited as context information.
- **VTransformer** [50]: A transformer based model which independently generates descriptions for each event segment. We use ground-truth event segments for a stronger baseline as in [13].
- **AdvInf** [21]: A LSTM based model with hybrid discriminators to select diverse and fluent captions from a sampling set at inference time. The ground-truth event segments are used to generate captions.
- **MART** [13]: A transformer based model with memory augmented to fully exploit the event and sentence histories for better captioning. Similar to AdvInf [21], the ground-truth event segments are used in generation.

Table 1 reports the paragraph generation performances of different models on the ActivityNet Captions ae-test split. The table shows that there is a large performance gap for two-stage approaches between using ground-truth event segments (Row 1-4) and using automatically generated ones (Row 5), which demonstrates that the poor performance of event detection seriously hinders the quality of paragraph generation. With the advantages of one-stage framework, vanilla model and our final model both outperform these strong baselines and do not require any event segment annotations. Our model achieves the best results on both the accuracy and diversity aspects except on METEOR. The METEOR metric prefers longer paragraphs, therefore, the methods using ground-truth event proposals have higher METEOR scores due to the same number of sentences with the ground-truth. For the diversity aspect, we achieve competitive results with the human level on Div@1 and Div@2, which demonstrates the effectiveness of our proposed dynamic video memories and diversity-driven training objectives. Experimental results on Charades Captions dataset in Table 2 also demonstrate our model achieves state-of-the-art results for video paragraph generation. Besides automatic metrics, we also conduct human evaluation in the supplementary material to further show the improvements.

4.3. Trade-off of Efficiency and Performance

Figure 2 shows the inference speed and captioning performances of our model using different keyframe selecting ratios. The inference speed grows (the time cost drops) rapidly with fewer video clips to be attended to, while the CIDEr drops slowly until the selecting ratio δ is below 0.4. It demonstrates that the proposed keyframe selection mechanism is effective which can discard less informative clips in the untrimmed video to improve efficiency and maintain the captioning quality. The best selecting ratio to trade off the speed and performance is 0.5. Contrary to the accuracy metric, the diversity of paragraph is improved with fewer frames selected because the video features are more distinctive without redundancy.

We also compare our proposed keyframe selection method with a uniform interval sampling approach, which
Table 3. Ablation study on ActivityNet ae-test set to demonstrate the effectiveness of different components. (pme: progressive memory exposure, omd: over-accessed memory decay, token: token penalty objective, \(r_{rlv}\): relevance reward, \(r_{div}\): phrase penalty objective.)

<table>
<thead>
<tr>
<th>#</th>
<th>Decoder</th>
<th>MLE</th>
<th>RL</th>
<th>BLEU@4</th>
<th>METEOR</th>
<th>CIDEr</th>
<th>Div@1↑</th>
<th>Div@2↑</th>
<th>Rep@4↓</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>11.53</td>
<td>15.91</td>
<td>24.11</td>
<td>64.92</td>
<td>82.34</td>
<td>3.17</td>
</tr>
<tr>
<td>2</td>
<td>✓</td>
<td></td>
<td></td>
<td>11.95</td>
<td>15.94</td>
<td>25.52</td>
<td>66.79</td>
<td>82.81</td>
<td>3.39</td>
</tr>
<tr>
<td>3</td>
<td>✓</td>
<td></td>
<td></td>
<td>11.91</td>
<td>16.01</td>
<td>24.47</td>
<td>66.18</td>
<td>82.95</td>
<td>2.87</td>
</tr>
<tr>
<td>4</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>11.61</td>
<td>15.72</td>
<td>25.65</td>
<td>67.90</td>
<td>83.37</td>
<td>2.80</td>
</tr>
<tr>
<td>5</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>11.74</td>
<td>15.64</td>
<td>26.55</td>
<td>68.42</td>
<td>83.95</td>
<td>2.75</td>
</tr>
<tr>
<td>6</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>12.10</td>
<td>15.85</td>
<td>27.06</td>
<td>67.81</td>
<td>83.45</td>
</tr>
<tr>
<td>7</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>12.20</td>
<td>16.10</td>
<td>27.36</td>
<td>68.33</td>
</tr>
</tbody>
</table>

Table 4. Ablation study on ActivityNet ae-test set to demonstrate contributions of different losses for keyframe selection with ratio \(\delta = 0.5\).

<table>
<thead>
<tr>
<th>#</th>
<th>(L_{\text{mle}})</th>
<th>(L_{\text{recons}})</th>
<th>(L_{\text{sparsity}})</th>
<th>BLEU@4</th>
<th>METEOR</th>
<th>CIDEr</th>
<th>Div@1↑</th>
<th>Div@2↑</th>
<th>Rep@4↓</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>✓</td>
<td></td>
<td></td>
<td>11.60</td>
<td>15.51</td>
<td>25.47</td>
<td>66.97</td>
<td>81.06</td>
<td>4.30</td>
</tr>
<tr>
<td>2</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>11.41</td>
<td>15.58</td>
<td>26.80</td>
<td>68.16</td>
<td>82.75</td>
<td>2.82</td>
</tr>
<tr>
<td>3</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>11.34</td>
<td>15.27</td>
<td>26.02</td>
<td>68.06</td>
<td>81.45</td>
<td>3.77</td>
</tr>
<tr>
<td>4</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>11.67</td>
<td>15.71</td>
<td>26.74</td>
<td>68.71</td>
<td>83.23</td>
</tr>
</tbody>
</table>

Figure 2. Variation of captioning performance and speed with different selecting ratios \(\delta\). TPV denotes time per video (ms).

Figure 3. Visualization of the attention weights \(\alpha\) on video frames for paragraph generation. *Left:* vanilla model. *Right:* our model with dynamic video memories.

uniformly selects video clip features instead of relying on the learned \(s^3\). Our proposed keyframe selection method outperforms the uniform interval sampling on both CIDEr and Rep@4 under all the selecting ratios, which demonstrates our model can select more distinctive frames.

### 4.4. Ablation Studies

In order to demonstrate the contributions from different components in our model, we carry out ablation studies in Table 3. Row 1 denotes the vanilla baseline presented in Section 2, which directly generates the video paragraph as in conventional video captioning task. In Row 2-4, we replace the standard attention mechanism in the vanilla transformer decoder with our proposed attention enhanced by dynamic video memories. The proposed progressive memory exposure and over-accessed memory decay mechanisms both improve the paragraph diversity by 1-2 points on Div@1, while combining them achieves additional gains on both accuracy and diversity metrics. We visualize the attention weights from vanilla model and our proposed model in Figure 3. It shows the learned attentions in vanilla model only focus on a few salient clips for the whole paragraph, which leads to repeated or missed event descriptions. However, with our proposed dynamic video memories according to the description status, our attention can focus on diverse frames and roughly forms a diagonal line similar to human description with the chronological order. In Row 5, we add the token-level high frequency penalty to the MLE training objective, which further brings improvements. In Row 6 and Row 7, we finetune the pretrained model via reinforcement learning. With the CIDEr reward alone, the model yields better accuracy results, but harms diversity metrics. Combining it with phrase-level high frequency penalty achieves the best final results. The proposed diversity-driven training objectives (both token- and phrase-level) are shown not only improving the diversity metrics but also the accuracy metrics due to their good abilities to prevent the language decoder from generating high-frequency words regardless of visual content.

Table 4 shows the effectiveness of the two auxiliary losses in the keyframe selection. We report the results with 50% video clips selected in inference since it is the best ratio choice to trade off the speed and performance. Without the video reconstruction loss, the model learns to select keyframes only with the captioning loss, which results in a large performance drop. However, the proposed recon-
struction loss can help to enforce the model to select salient frames that maintain similar semantic information with the original video clip features. Our final loss with the three losses combined achieves the best results.

4.5. Qualitative Analysis

Figure 4 shows a test example with the paragraph captions from our model and other state-of-the-art models. The compared models incorrectly describe the woman as “man” due to its higher frequency (man 2.26% vs. woman 0.85%) in the data. Furthermore, they tend to generate redundant high-frequency phrases regardless of the video content, such as “speaking to the camera”, which is the top1 frequent verb phrase in the training set. Our model, however, can generate more coherent and diverse video paragraphs even without ground-truth event segment annotations. More examples can be found in the supplementary material.

5. Related Works

Over the past years, image captioning has achieved significant improvements [32, 41, 43, 15, 1], which mainly focus on generating a single sentence to describe the image content. In order to describe more fine-grained details in an image, Krause et al. [11] propose the image paragraph captioning with a hierarchical RNN to generate topic vectors first and then convert topics into sentences to form a paragraph. However, recent works [16, 17] have shown that directly generating paragraph as a long sentence outperforms the hierarchical manner when enhanced with diversity-driven training and inference approaches. Inspired by the image paragraph generation pioneers, in this work we explore whether the video paragraph can be effectively generated without hierarchical manner of using event detection.

Video captioning [31, 44, 35, 5, 33] is more challenging compared to image captioning with complexities on both spatial and temporal dimensions. Recently, Krishna et al. [12] propose the dense video captioning task to localize and describe multiple events for long videos with multiple sentences. They first detect multiple events in the video and then generate description for each of them. However, these descriptions are independent and not coherent as a whole. Xiong et al. [39] further propose to generate a coherent paragraph to describe multiple activities in the long video. However, they still solve the problem in a two-stage way like dense video captioning methods [50, 34, 14, 18]. They first propose hundreds of event proposals with event proposal networks [48, 2, 8], then select events to be described with the contextual information from previously detected events and generated captions. Park et al. [21] and Lei et al. [13] directly generate video paragraphs with the ground-truth event segments, which cannot generalize to videos without such event annotations. In this work, we eschew the event detection stage and directly generate the video paragraph with dynamic video memories.

6. Conclusion

In this paper, we propose an one-stage framework for video paragraph generation. Due to the long video inputs and paragraph outputs, it is challenging to generate diverse paragraphs efficiently. We propose a keyframe-aware video encoder to improve the efficiency and an attention mechanism with dynamic video memories to learn more diverse and coherent visual attentions. Besides, a diversity-driven training objective with high-frequency token and phrase penalties is proposed to improve language diversity. Experimental results on ActivityNet and Charades datasets show that our proposed model outperforms the state-of-the-art performance on both accuracy and diversity metrics.

Acknowledgment

This work was supported by National Natural Science Foundation of China (No. 62072462), Beijing Natural Science Foundation (No. 4192028), and National Key R&D Program of China (No. 2020AAA0108600).
References


